'IUSSU' ADMINISTRADOR DE COLECCIONES

ÁNGELA VIRGINIA AMARÍS IBÁÑEZ

ANGIE MILENA PINEDA CASALLAS

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INFORMÁTICA Y ELECTRÓNICA
PROGRAMA DE TECNOLOGÍA EN INFORMÁTICA
BOGOTÁ D.C.
2007

'IUSSU' ADMINISTRADOR DE COLECCIONES

ÁNGELA VIRGINIA AMARÍS IBÁÑEZ

ANGIE MILENA PINEDA CASALLAS

Trabajo de Grado para optar al Título
de Tecnólogo en Informática

Asesor
Luis Eduardo Perez

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INFORMÁTICA Y ELECTRÓNICA
PROGRAMA DE TECNOLOGÍA EN INFORMÁTICA
BOGOTÁ D.C.
2007

CONTENIDO

	pág.
INTRODUCCIÓN	9
1. PROYECTO 'IUSSU'	11
1.1 TEMA	11
1.2 TÍTULO DEL PROYECTO	12
1.3 PLANTEAMIENTO DEL PROBLEMA	13
1.4 DESCRIPCIÓN DEL PROBLEMA	14
1.5 JUSTIFICACION	15
1.6 OBJETIVOS	16
1.6.1 Objetivo General	16
1.6.2 Objetivos Específicos	17
1.7 METODOLOGÍA DE INVESTIGACIÓN	17
1.8 LÍNEA DE INVESTIGACIÓN	19
2. MARCO REFERENCIAL	21
2.1 ESTADO DEL ARTE	21
2.2 FUNDAMENTOS TEÓRICOS	26
2.3 MARCO LEGAL	29
3. METODOLOGÍA DE DESARROLLO	32
4. ANÁLISIS Y DISEÑO	39
4.1 DEFINICIÓN DEL SISTEMA PROPUESTO	39
4.2 DESCRIPCIÓN DETALLADA DEL SISTEMA PROPUESTO	40

4.3 DESCRIPCIÓN DETALLADA DE CADA PROCESO 44

4.3.1 Casos de Uso del Proyecto IUSSU 44

4.3.2 Flujo de Páginas Web (Diseño del Sistema Propuesto) 49

4.4 DICCIONARIO DE DATOS 51

4.5 MODELO ENTIDAD RELACIÓN 56

5. DESARROLLO 57

5.1 ESPECIFICACIONES TÉCNICAS 57

5.1.1 Software 57

5.1.2 Hardware 58

6. GLOSARIO 59

7. CONCLUSIONES 63

BIBLIOGRAFÍA 66

ANEXOS 69

LISTA DE TABLAS

pág.

Tabla 1: Listado de Frameworks por Lenguaje de Programación 18

Tabla 2: Prácticas y Descripción de Metodologías Ágiles. 38

Tabla 3: Formato para Establecer la Visión del Proyecto 40

Tabla 4: Reserva del Producto. 44

Tabla 5: Tipos de Datos en Rails 51

Tabla 6: Tabla de Usuarios 52

Tabla 7: Tabla de Colecciones 53

Tabla 8: Tabla de Canciones 54

Tabla 9: Tabla de Documentos 54

Tabla 10: Tabla de Imágenes 55

Tabla 11: Tabla de Otros 55

LISTA DE FIGURAS

pág.

Figura 1. Screenshot Yahoo Music. 22

Figura 2. Screenshot Napster. 23

Figura 3. Screenshot Alexandria. 25

Figura 4. Screenshot Tellico 26

Figura 5: Flujo de Páginas Web del Usuario 50

Figura 6: Flujo de Páginas Web del Usuario Detallado 50

LISTA DE DIAGRAMAS

pág.

Diagrama 1. Arquitectura Modelo-Vista-Controlador en Rails 27

Diagrama 2. Esquema de Funcionamiento de Ruby on Rails 29

Diagrama 3: Caso de Uso Registro 45

Diagrama 4: Caso de Uso Login 45

Diagrama 5: Caso de Uso Agregar Nueva Colección 46

Diagrama 6: Caso de Uso Agregar Item a Colección 47

Diagrama 7: Caso de Uso Reproducir/Visualizar Item 48

Diagrama 8: Caso de Uso Administración de Usuarios 49

Diagrama 9: Modelo Entidad-Relación 56

LISTADO DE ANEXOS

pág.

ANEXO A GNU GENERAL PUBLIC LICENSE 69

ANEXO B RUBY LICENSE 77

ANEXO C THE MIT LICENSE 79

ANEXO D BSD LICENSE LIGHTTPD 80

ANEXO E CREATIVE COMMONS ATTRIBUTION 2.5 82

ANEXO F MANUAL DE USUARIO 89

ANEXO G MANUAL DEL SISTEMA 103

0. INTRODUCCIÓN

En nuestro idioma se hace una distinción entre el mirar y el ver. Mirar sería la acción en la que obtenemos la información del mundo a través de nuestros sensores ópticos, mientras que ver implica un procesamiento de dicha información.

Haciendo una analogía de la visión con los medios para obtener información y su procesamiento. Podemos darnos cuenta de que en nuestro mundo, obtener información no es un problema. Tanto física como digitalmente un individuo promedio tiene acceso a millones de fuentes de consulta cualquiera sea su área de interés. Y de dicha información filtrada por las limitaciones de las interfaces humanas (educación, memoria, disposición, tiempo...) nuestros cerebros extraerán finalmente la información que les es útil.

El desafío no es pues obtener la información, sino filtrarla de forma que pueda ser útil, sometiéndola a procesos de clasificación, comparación, organización y selección.

Dentro de este amplio marco, que es el procesamiento de la información, el presente documento plantea un proyecto para Administrar Colecciones. Una herramienta para que los usuarios de dicho sistema puedan llevar un registro de los datos de sus colecciones bien sean estas música, imágenes o documentos a los que tiene acceso, pero sobre todo, una herramienta para 'ver' sus Colecciones, en todo el sentido del término.

La meta de este proyecto final del programa de Tecnología en Informática, es el

desarrollo de software, la realización de una aplicación práctica en la que se empleen las competencias adquiridas durante la carrera.

También se pretende investigar por medio de este trabajo la utilización de nuevas herramientas Open Source que aceleran el proceso de desarrollo como un framework.

1. PROYECTO 'IUSSU'

1.1 TEMA

Comenzar un proyecto de desarrollo de software implica tomar múltiples decisiones que repercutirán directamente sobre la calidad de la aplicación que se desea desarrollar. Así que el primer tema de un proyecto de desarrollo de software es la investigación sobre las tecnologías y los recursos disponibles para llegar a obtener el producto deseado.

¿Cumple la aplicación con las funcionalidades para las que fue diseñada? ¿Qué tan difícil de ser transferida a otro entorno es?¿La información que maneja persiste de forma consistente y confiable?¿Es fácil de utilizar?¿Es eficiente?¿Qué tan difícil es realizar una nueva modificación?.

No existen fórmulas mágicas de diseño o planificación, ni un sólo lenguaje de programación que se ajuste a todas las necesidades, así que las decisiones se deben tomar de acuerdo a cada situación concreta.

El tema de éste proyecto, planteado en estos términos, es una investigación sobre las ventajas y desventajas de las Aplicaciones Web 2, escritas en el lenguaje de programación *Ruby*[1] , implementadas en el framework *Ruby on Rails*[2] .

Una Aplicación Web es en la que por medio del navegador los usuarios acceden a la información de un servidor remoto o local. Es una variación del patrón

1 Pagina Oficial Ruby A Programmer's Best Friend [http://www.ruby-lang.org/en/].

arquitectónico cliente-servidor tradicional, en el que cada instancia del cliente llama servicios sobre el servidor. Los navegadores funcionan como clientes ligeros y gracias al alto nivel de estandarización que rige a los navegadores se puede esperar un alto índice de compatibilidad entre diferentes sistemas operativos.

La Web 2.0 es la simbolización de la evolución de las aplicaciones tradicionales de escritorio hacia aplicaciones que generan colaboración y servicios Web enfocadas al usuario final .

Ruby por su parte es un lenguaje de programación interpretado y orientado a objetos creado por Yukihiro Matsumoto en 1993.

Un framework es un conjunto de componentes (generalmente clases y archivos de configuración) que aceleran el proceso de desarrollo al permitir reutilizar código ya existente que sigue modelos y patrones estandarizados de programación. En cierta forma se puede considerar como una aplicación genérica incompleta a la que se le pueden configurar y extender partes para crear una aplicación concreta.

Ruby on Rails es un framework escrito por David Heinemeier Hansson liberado al público por primera vez en Julio de 2004.

1.2 TÍTULO DEL PROYECTO

Para iniciar una búsqueda, se necesita una palabra, en cuanto se conoce, las asociaciones comienzan a surgir porque no hay una palabra, que no lleve a otra. Elegir el título de un proyecto es una suerte de invocación, en la que se reúnen por una parte las intenciones del proyecto y por otra la dirección hacia la que el

2 Pagina Oficial Ruby on *Rails*. Web development that doesn't hurt [http://www.rubyonrails.org/]

proyecto aspira a encaminarse.

El nombre de la aplicación a desarrollar como proyecto de grado es IUSSU[3], palabra en latín cuya traducción aproximada sería 'Por orden de'.

Así ésta aplicación aspira a que las colecciones obedezcan a sus usuarios. El resto del título del proyecto, 'Administrador de Colecciones' es una descripción de la principal utilidad de la aplicación, la creación, consulta, edición y eliminación de colecciones.

1.3 PLANTEAMIENTO DEL PROBLEMA

El conocimiento se construye a partir del conocimiento, como una gran función recursiva. Como diría con modestia Jorge Luis Borges “Soy todos los autores que he leído, toda la gente que he conocido, todas las mujeres que he amado, todas las ciudades que he visto, todos mis antepasados...”

Por desgracia, o por fortuna dirán algunos, las mentes humanas tienden inevitablemente hacia el olvido. Para una persona puede ser difícil saber con seguridad cuántos elementos constituyen una de sus colecciones. Por ejemplo, si se tratase de una colección de libros en formato digital, un usuario promedio tendría dificultades recordando cuántos ha leído, cuáles posee, el tema principal de cada uno de ellos, en qué formato se encuentran, el tamaño de cada uno de ellos y muy seguramente no los llevaría consigo a todo lado. ¿Cómo evitar que toda esta información se pierda? ¿Cómo hacer para que ésta información esté disponible para el usuario aunque éste no posea el software para reproducirlo o los permisos de administrador para instalar el software requerido para tal fin?

3 *IUSSU* -m n.: por orden de, al comando de, orden, mandato.

¿Cómo tener acceso a su colección, sin tener que instalar más cosas que un navegador de internet?

1.4 DESCRIPCIÓN DEL PROBLEMA

Durante la realización de cualquier tipo de colección, es muy común que el usuario sienta la necesidad de clasificarla, expresar su opinión personal sobre cada uno de los ítems que la constituyen y por qué no de compartir esta información con otras personas.

Así mismo dependiendo del tipo de colección también puede ser muy importante para su usuario el ubicar rápidamente uno de dichos ítems o reproducirlo por el medio adecuado, pero como podemos observar en diferentes sitios, ya sean de trabajo o de estudio es prohibido descargar cualquier aplicación de entretenimiento así sea de juegos, videos, MSN y música, sin contar con el hecho que algunos computadores no cuentan con el software adecuado de entretenimiento en música como Winamp, Windows media, etc. Por que estas aplicaciones necesitan de cierta memoria y su tamaño no permite que el computador de su mejor rendimiento.

Aunque la solución de muchos usuarios es burlar al administrador del lugar en el que se encuentran, y descargar la aplicación ésta no tarda en ser descubierta, y eliminada del PC, sin contar con los problemas que esto le puede ocasionar al usuario.

Tenemos así dos claras dificultades que hoy en día se presentan, por una parte está la rápida localización de un elemento en particular y por otro su reproducción. ¿Cómo hacer para que la posibilidad tecnológica de una aplicación informática con su agilidad en la manipulación de datos se utilice de forma productiva en un

proceso privado e individual?

1.5 JUSTIFICACION

Si bien es imposible recordarlo todo, saber dónde comenzar a buscar puede ser un muy importante primer paso. Empresas comerciales como Google, Yahoo, Amazon son por definición 'buscadores'. Han recopilado, centralizado, organizado, clasificando la información para la posterior consulta de sus usuarios.

Si bien crear colecciones es una actividad placentera para muchos, también es una actividad demandante de tiempo, memoria y comprensión. Recursos que cualquier persona valora y que por tanto no deberían desperdiciarse inútilmente.

Así, la información que una persona ha recopilado con su colección y que mentalmente ha organizado y clasificado, merece ser conservada de alguna forma no perecedera y confiable, en la que el acceso a esa información se pueda hacer posteriormente de una forma rápida .

Si se considera el problema de la instalación de software adicional y las restricciones que las empresas y universidades normalmente poseen, podemos tomar como ejemplo las aplicaciones de mensajería y chat como messenger y la solución que a dicho inconveniente brindan páginas como www.meebo.com[4] o el chat incorporado de gmail, que con ciertas limitantes al servicio tradicional, permiten que sus usuarios tengan los servicios básicos incorporados en una sola página web.

Una aplicación informática web puede informarle a un usuario, con exactitud los

4 meebo un programa de mensajería instantánea que se ejecuta dentro de un navegador web.

datos de un archivo. Informarle de su cantidad, su estado e incluso la posibilidad dependiendo del tipo de colección, de visualizarla o reproducirla de alguna forma, utilizando las características de los navegadores web.

Pero adicionalmente puede ayudarle a implementar sus propias categorías de clasificación por medio de etiquetas, ya que las clasificaciones tienden a ser arbitrarias, subjetivas y muchísimas veces imprevisibles desde el momento de su planteamiento.

Tenemos así que el administrador de colecciones solucionaría la lentitud en la búsqueda de los datos, integraría a la colección y la opinión del usuario de la misma, evitaría que mucha información se perdiese por el temible olvido y la imposibilidad de acceder a dicha información de forma oportuna y facilitaría en ciertos casos la visualización y reproducción de la misma.

1.6 OBJETIVOS

Las palabras objetivo y proyecto comparte una misma raíz latina acto que significa echar, arrojar, lanzar. El prefijo pro tiene el significado de ante o delante mientras que el prefijo ob significa a causa de. Se tiene así que si un proyecto es arrojar algo hacia adelante, sus objetivos son las causas de dicha acción.

1.6.1 Objetivo General

Adquirir destrezas en el desarrollo de aplicaciones web que almacenen la información de forma persistente en una base de datos escritas en el lenguaje *Ruby* implementadas en el framework *Ruby on Rails*.

1.6.2 Objetivos Específicos

Desarrollar una Aplicación Web que permita a sus usuarios reproducir y visualizar documentos de texto, música e imágen.
Agilizar la gestión de colecciones que cada usuario posee disminuyendo los tiempos de búsqueda y la pérdida de información.

Permitir el acceso de cada usuario a la información de sus colecciones de una forma segura, independientemente del lugar en que se encuentre.

1.7 METODOLOGÍA DE INVESTIGACIÓN

Una vez planteada la utilidad del proyecto, es necesario que se defina la profundidad con la que se espera abordarlo así como el modo de llevarlo acabo.

En el presente uno de los enfoques más prometedores en los nuevos entornos de desarrollo es el diseño de arquitecturas de software que permitan la reutilización de código por medio de la construcción de aplicaciones composicionales, en las que el usuario pueda concentrarse en lidiar con los requerimientos del software en lugar de los mas tediosos detalles del proceso de programación. Cada año aumenta el número de frameworks multipropósito, construidos en los más populares lenguajes de programación (Ver Tabla 1). Sin embargo este tipo de desarrollo no hace parte de la formación académica de la mayoría de universidades colombianas, bien por su novedad y variedad o bien por que éste tipo de temas se extiende más allá de los contenidos de su pensum.

Lenguaje de Programación	Framework
Java	Apache Struts
	Hibernate
	JavaServer Faces
	CNI
	Java Native Interface
PHP	Code Igniter
	Symfony
	Zend Framework
	cakePHP
Python	CherryPy
	Django
	TurboGears
Ruby	Ruby on Rails
	Nitro

Tabla 1: Listado de Frameworks por Lenguaje de Programación

Con el panorama así planteado, el Proyecto IUSSU se encuentra en el nivel de estudio Exploratorio, ya que tiene pocos antecedentes en cuanto a su modelo práctico en el ámbito académico. No se cuenta con la asesoría de ningún experto, ni con experiencia previa en el lenguaje de programación y framework escogidos pero sí con la documentación que al respecto se encuentra en internet (especialmente en el idioma inglés).

Según Mario Tamayo Tamayo en el segundo Módulo de La Investigación de la Serie Aprender a Investigar, existen tres tipos de investigación principales, la Investigación Histórica centrada en la descripción de lo que era, la Investigación Descriptiva centrada en la descripción de lo que es y la Investigación Experimental

centrada en la descripción de lo que será. El Proyecto *IUSSU* no está interesado en comprobar explicaciones, ni en probar determinadas hipótesis, ni en hacer predicciones, sino en describir de la forma más claramente posible el proceso de desarrollo de software que implementa frameworks de forma que pueda servir de base para la realización de nuevas investigaciones por otros autores, por lo que el tipo de investigación que más se ajusta al proyecto es la Investigación Descriptiva.

Dado que la totalidad de las fuentes de información se encuentran en internet, el proceso de recolección de la información para llevar a cabo el proyecto, estará iterando constantemente a lo largo del proceso de desarrollo entre la consulta de dicha información y la implementación paulatina del material investigado.

1.8 LÍNEA DE INVESTIGACIÓN

UNIMINUTO se propone dar una formación integral a sus estudiantes tanto desde la producción de conocimiento, como desde la praxis, la transformación de sus conocimientos en comportamientos y actitudes que repercutan de forma positiva en la comunidad. Visto de esta forma, la investigación como medio de organizar la experiencia del trabajo académico es importante y la forma de consolidarla para UNIMINUTO, es a través del establecimiento de líneas de investigación.

Estas son, la línea de Innovación educativa y transformación social, la línea de lenguaje, comunicación y pensamiento, la línea de innovaciones tecnológicas y cambio social y la línea de gestión, participación y desarrollo comunitario.

Dentro de las líneas de investigación que propone UNIMINUTO, la Facultad de Ingeniería se inscribe en la línea de Innovaciones tecnológicas y cambio social. A su vez, el programa de Tecnología en Informática, ofrece cuatro sublíneas con las

que se compromete con la institución para generar políticas que orienten a sus estudiantes a realizar investigación aplicada. Estas son Desarrollo de software, Sistemas de Información, Redes de Computadores y Plataformas.

El Presente proyecto si bien tiene características con las que podría inscribirse en varias sublíneas, como:

Plataformas ya que los frameworks pueden considerarse plataformas al ser ya una solución informática parcialmente configurada.

Redes de Computadores, ya que la arquitectura en la que descansa el proyecto es la tradicional cliente-servidor y el hecho de que El Proyecto *IUSSU* esté pensado como un servicio web y no como una aplicación final de escritorio tradicional.

O Sistemas de Información ya que la principal utilidad de la aplicación es procesar entradas, mantener archivos de datos y producir información útil para su usuario.

Sin embargo ha sido inscrito en la sublínea de Desarrollo de Software ya que es la que describe más fielmente el objetivo de *IUSSU* que es el desarrollo de una aplicación codificada en el lenguaje de programación *Ruby* que utiliza el framework *Ruby on Rails*.

2. MARCO REFERENCIAL

Con el propósito de ampliar el panorama en el que se encuentra inscrito el desarrollo de software de una Aplicación Web, como *IUSSU*, dentro del Marco Referencial se proseguirá a exponer, el Estado del Arte, los Fundamentos Teóricos y el Marco Legal de los lenguajes, tecnologías y aplicaciones que se requieren para su implementación.

2.1 ESTADO DEL ARTE

Actualmente existen diferentes aplicaciones on-line y aplicaciones de escritorio que prestan servicios similares al propuesto por el proyecto *IUSSU*. Entre las aplicaciones on-line se destacan *Yahoo music* y *Napster* siendo su función principal la de reproducir colecciones de música. Y entre las aplicaciones de escritorio *Alexandría* y *Tellico* definidas como Administradores de Colecciones principalmente de libros. A continuación se presenta una breve reseña de cada una de estas aplicaciones mencionadas anteriormente:

- *Yahoo music*[5] Es una de las aplicaciones que reproduce música en Internet, ésta aplicación permite reproducir música que se acaba de comprar o que tiene el usuario a su disposición (sólo si es legal). Funciona a través de un portal similar a un mercado libre que permite oír la música que se va a comprar, y de cierta forma comprometiendo al comprador potencial.

  Si el usuario no compra, la página o el servidor empieza a enviar toda clase de promoción para que el usuario compre.

5 Yahoo! Iberia, S.L. [en línea] http://es.launch.yahoo.com 2007 [citado el 02-22-07]

Es indispensable que el usuario se inscriba a este, con información general, claro esta que si llega a pagar un"pequeño" costo recibirá información de los últimos tracks (canciones) de música y reproducir ciertas cosas tales como Bonus track o canciones que aún no han salido a la venta pero son promocionales.

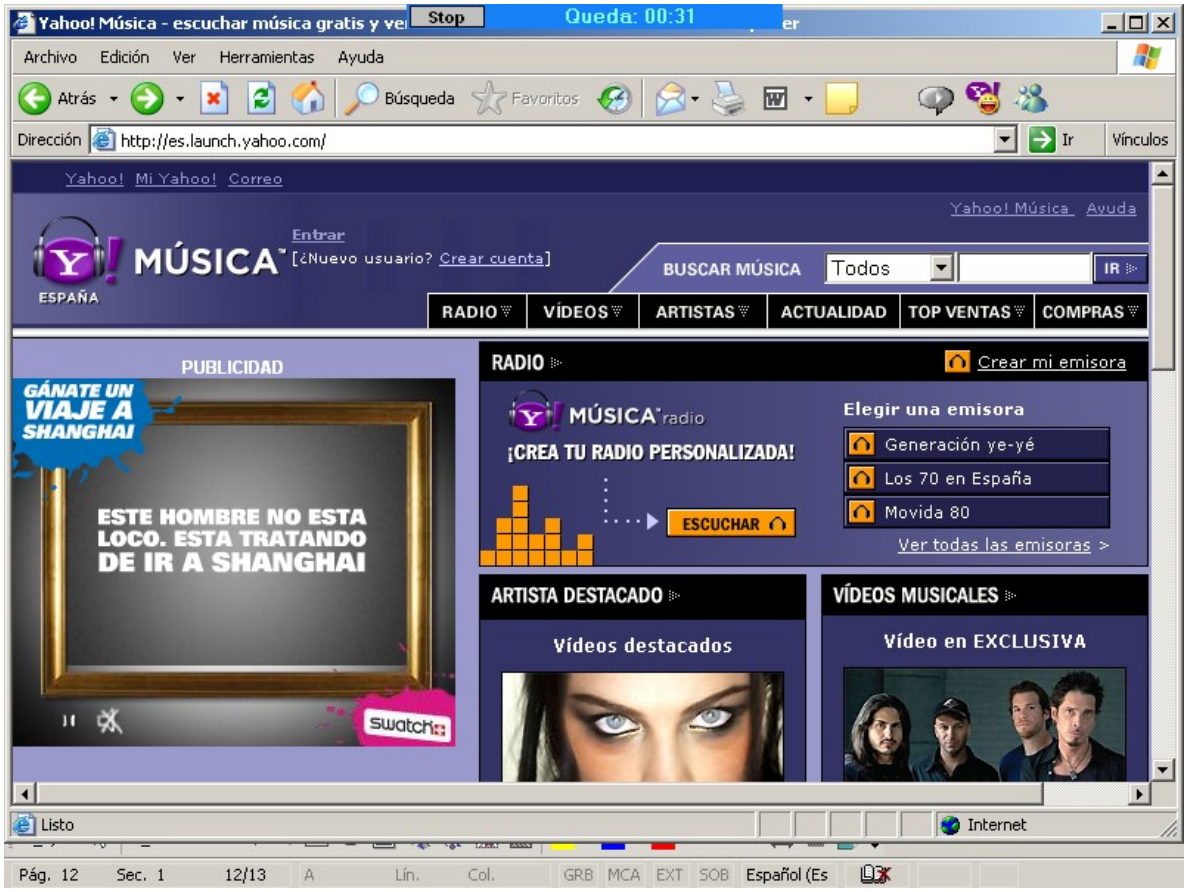


Figura 1. Screenshot Yahoo Music.

- *Napster*[6] Otro tipo de reproductor on line, famoso porque antiguamente permitía descargar la música sin restricciones, pero a raíz de una demanda multimillonaria con Metallica (Grupo de Metal), perdieron mucho dinero. Y para no ser vetados se convirtieron en defensores de los derechos de autor.

6 Napster, LLC. Napster. [en línea] http://www.napster.com 2003-2007 [citado el 02-22-07]

Napster permite buscar el track ya sea por el artista, álbum o por alguna palabra clave.

Después de elegida en la parte inferior le muestra al usuario, la procedencia de la canción y da el número que ocupa la canción en la base de datos obviamente para ser descargado bajo un precio o para reproducir una muestra de la canción.

La gran desventaja es que el usuario finalmente oye la canción que haya elegido, pero no puede reproducir un listado de canciones, es decir cuando acabe la canción debe devolverse, buscar nuevamente, y reproducir la que él elija.

El portal no siempre reproduce la canción elegida por lo que él aconseja que se descargue el reproductor, y se descargue la canción.

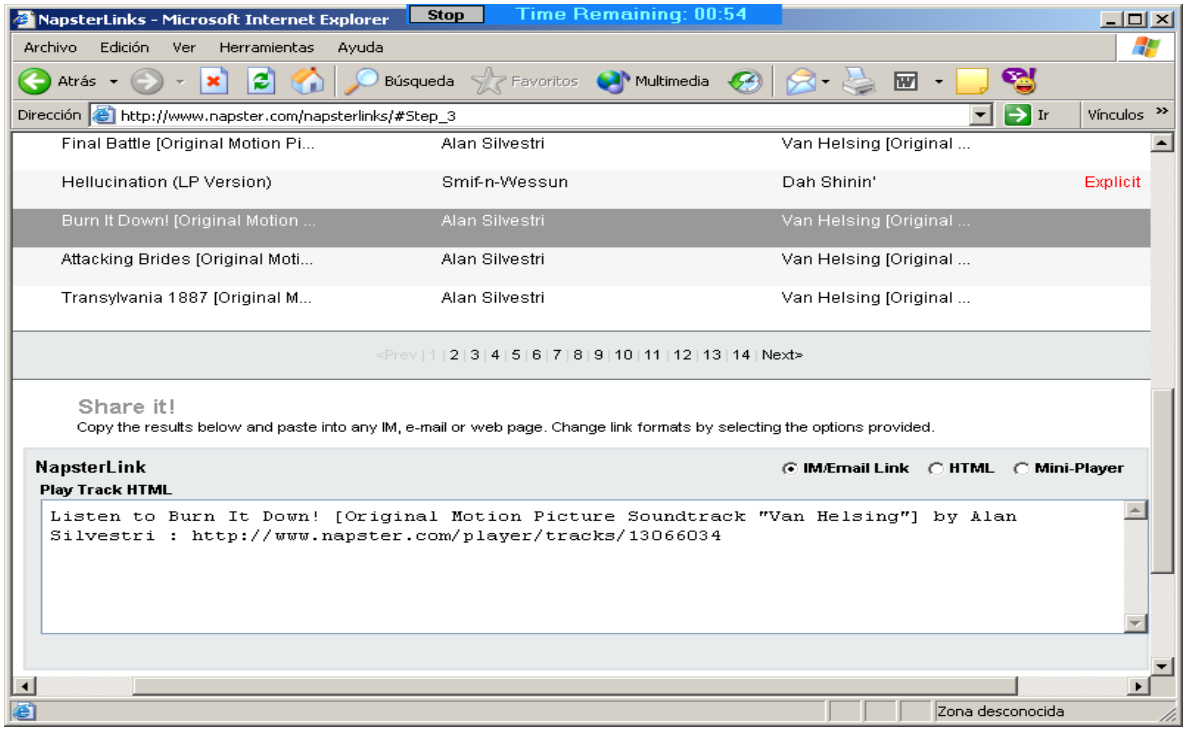

NapsterLinks - Microsoft Internet Explorer
Stop
Time Remaining: 00:54
Archivo Edición Ver Herramientas Ayuda
Atrás
Búsqueda
Favoritos
Multimedia
Dirección
http://www.napster.com/napsterlinks/#Step_3
Ir
Vínculos
Final Battle [Original Motion Pi... Alan Silvestri Van Helsing [Original ...
Hellucination (LP Version) Smif-n-Wessun Dah Shinin' Explicit
Burn It Down! [Original Motion ... Alan Silvestri Van Helsing [Original ...
Attacking Brides [Original Moti... Alan Silvestri Van Helsing [Original ...
Transylvania 1887 [Original M... Alan Silvestri Van Helsing [Original ...
<Prev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | Next>
Share it!
Copy the results below and paste into any IM, e-mail or web page. Change link formats by selecting the options provided.
NapsterLink
IM/Email Link
HTML
Mini-Player
Play Track HTML
Listen to Burn It Down! [Original Motion Picture Soundtrack "Van Helsing"] by Alan Silvestri : http://www.napster.com/player/tracks/13066034
Zona desconocida

Figura 2. Screenshot Napster.

Dentro del marco del Software Libre, se destacan dos aplicaciones cuyo objetivo es el manejo de Colecciones de Libros. Estas son los proyectos Alexandria y Tellico. Estas aplicaciones, si bien son software libre, no son multiplataforma. Y las observaciones que el usuario puede hacer sobre cada item son generales, muy al estilo de un producto anunciado en un portal comercial como Amazon.com.

- *Alexandría*[7] es una aplicación GNOME escrita en *Ruby* con una interfaz muy básica para administrar colecciones de libros, permite describir bibliograficamente cada libro, asociar la imagen de su carátula, agregar una breve nota personal por libro y una calificación. Realiza diferentes tipos de búsquedas por nombre, autor e ibns y permite exportar la colección como una pagina web.

7 SANSONETTI, Laurent. Alexandria [en linea] http://alexandria.rubyforge.org 2004-03-20 [citado el 27 de Mayo 2006]

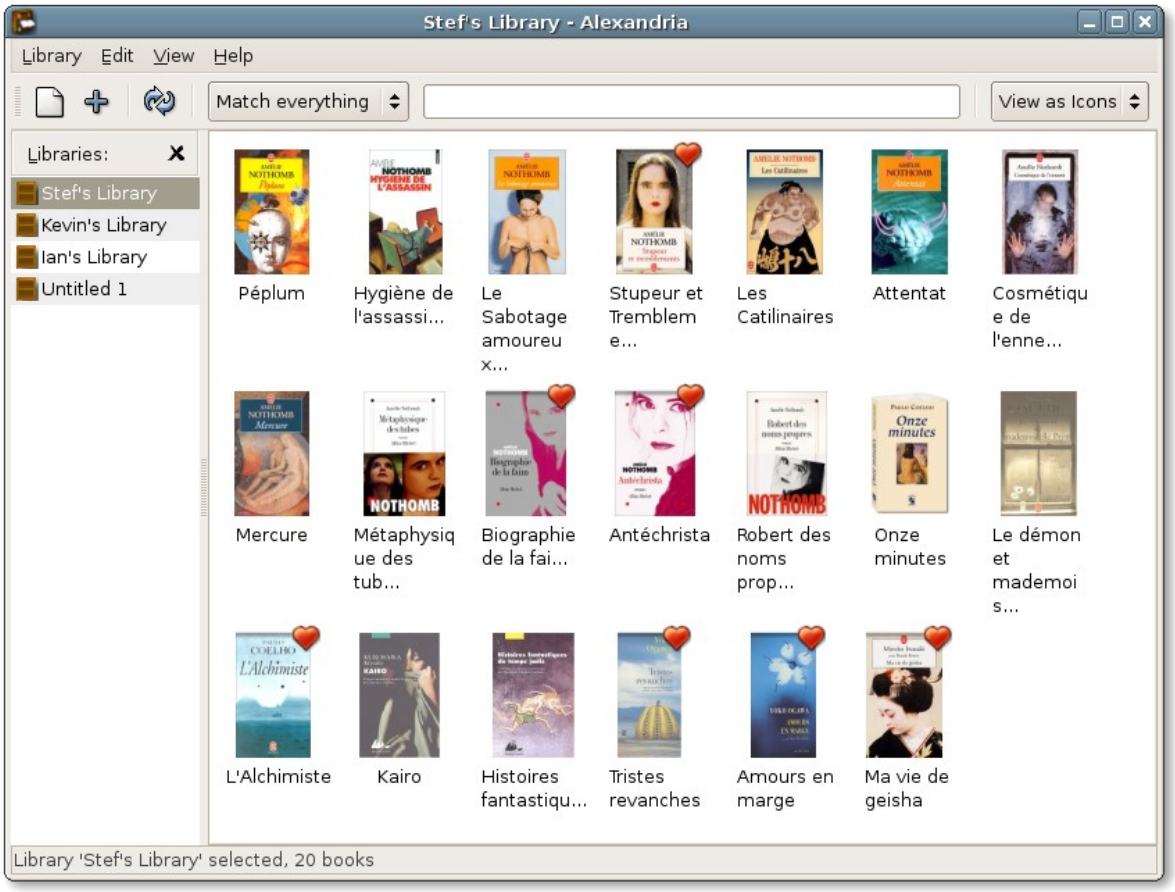


Figura 3. Screenshot Alexandria.

- *Tellico*[8] es una aplicación KDE escrita en C++ con una interfaz muy amigable para administrar colecciones no solo de libros, sino de comics, videos, música, monedas, estampillas, postales y vinos. Permite describir las principales características de cada elemento de cada tipo de colección, asociar imagenes, agregar una breve nota personal por artículo así como su calificación. Realiza diferentes tipos de búsquedas por criterios definidos por el usuario.

8 STEPHENSON Robby. Tellico A collection manager for KDE 3.x. [En linea] http://www.periapsis.org/tellico/ 2002-2006 [citado el 27 de Mayo 2006]

Figura 4. Screenshot Tellico

A diferencia de las anteriores aplicaciones on-line el Proyecto *IUSSU* no tiene como objetivo, vender música ni promocionar nuevos artistas, sino servir como espacio de almacenamiento para el material legal que el usuario ya posee. Y a diferencia de las anteriores aplicaciones de escritorio, el Proyecto *IUSSU* es multiplataforma y el usuario puede tener acceso a su información sin importar el lugar en donde se encuentre.

2.2 FUNDAMENTOS TEÓRICOS

En primera instancia se expone el funcionamiento interno del framework *Ruby on Rails*, la arquitectura que lo sustenta así como el esquema de funcionamiento típico de una aplicación *Rails*.

Arquitectura *Ruby on Rails*

Ruby on Rails es un framework, open source escrito en *Ruby* que sigue de cerca el Patrón de Arquitectura Modelo-Vista-Controlador (MVC) formulado por primera vez en 1979 por Trygve Reenskaug un científico de computación noruego quien trabajaba para los laboratorios de investigación de Xerox en Smalltalk.

Él patrón MVC de arquitectura consta de tres partes:

El modelo: que es el responsable por mantener el estado de la aplicación, los datos y las reglas que aplican sobre los datos.

La vista: que es responsable por generar la interfaz para el usuario, normalmente basándose en el modelo de datos.

El/los Controlador/es: que dirigen la aplicación, reciben eventos, interactúan y despliegan la vista apropiada para el usuario.

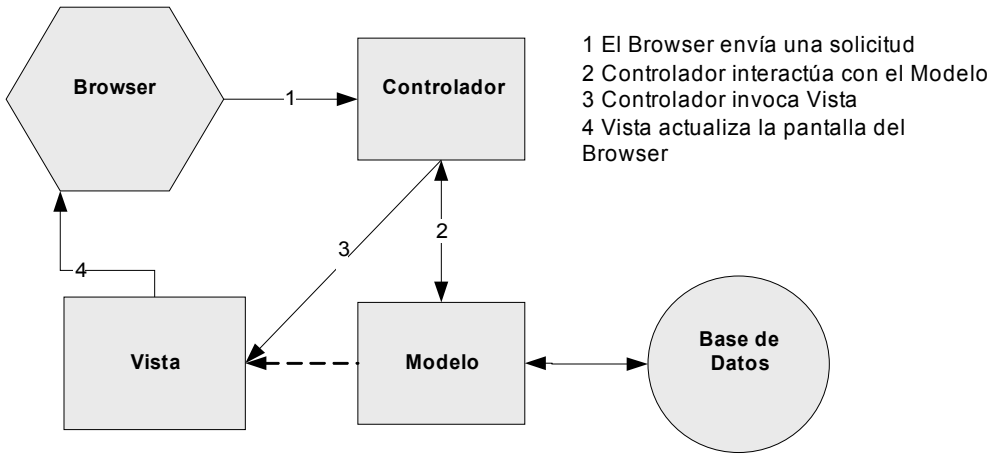


Diagrama 1. Arquitectura Modelo-Vista-Controlador en Rails

Una aplicación *Rails* es contenida en un servidor. Para desarrollo se suele emplear WEBrick que es un servidor cien por ciento *Ruby* que viene incluido con el lenguaje desde la versión 1.8.1 y posteriores. Sin embargo para uso comercial se recomienda la utilización de otros servidores como Apache o Lighttpd.

Cuando una petición es hecha desde el navegador, ésta es enviada en primer lugar hacia un router el DispatchServlet que descomponiendo la dirección URL de donde se produjo el evento determina la instancia del controlador o el método que debe invocar para producir la acción deseada.

Una acción es simplemente una clase que extiende de la clase ActionController. De ser necesario, ésta invoca los cambios en el modelo y/o en la vista.

El modelo es otra clase que extiende un comportamiento por defecto ésta vez de ActiveRecord::Base. Cuando es necesario, ésta clase es la encargada de hacer persistir la información en la base de datos utilizando un modelo de dos capas de driver-adaptador. El driver es la librería de *Ruby* que se conecta con la base de datos (DB2, Firebird, MySQL, Oracle, Postgres, SQLServer o SQLite).

La actualización de la vista se lleva a cabo por medio de RHTML, HTML con *Ruby* embebido y CSS hojas de estilo. Un esquema de todo este proceso puede verse en el Diagrama número 2.

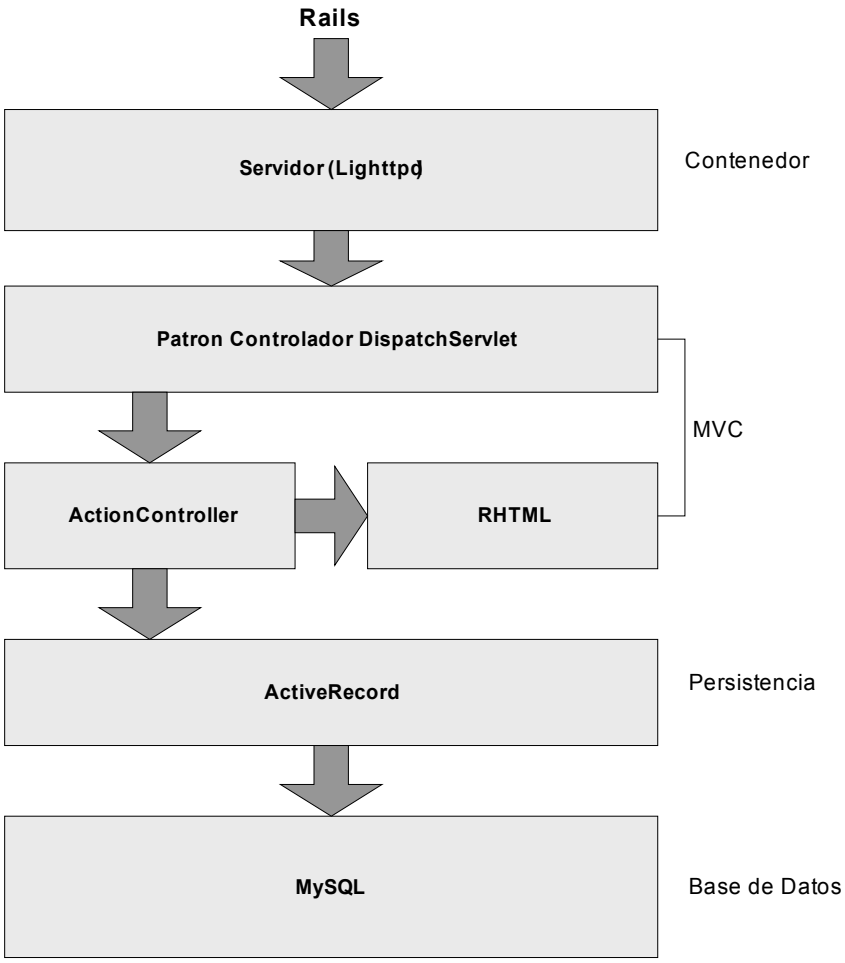


Diagrama 2. Esquema de Funcionamiento de Ruby on Rails

2.3 MARCO LEGAL

Con el propósito de establecer el uso permitido de las tecnologías y aplicaciones que se utilizan en el desarrollo de la aplicación *IUSSU*, se explicarán las principales características de sus licencias. Una copia de cada una de las licencias podrá consultarse en los Anexos.

El lenguaje de programación *Ruby* fue liberado bajo la GPL[9] (General Public License) la licencia del Software Libre o una versión específica llamada The *Ruby* License[10].

Ruby on Rails fue liberado bajo la licencia Open Source. MIT License[11] (Masachusetts Institue of Technology).

La aplicación servidor funciona sobre Fedora Core 6. Un sistema operativo de propósito general basado en Linux, desarrollado por un gran número de personas que se esfuerzan en proveer y mantener lo mejor del software libre, el software de código abierto y estándares libres. Patrocinado por Red Hat, Inc.

El servidor web que soporta la aplicación es Lighttpd, que es software libre y se distribuye bajo la licencia BSD[12] (Berkeley Software Distribution).

Los plugins acts-as-taggable-on-steroids, file-column, smtp-plugin, validates-date-time son Open Source y se distribuyen bajo la licencia MIT

El javascript Lightbox usado para exhibir imágenes se distribuye con la Licencia Creative Commons[13] de la atribución 2.5 de los campos comunes.

Zoho[14] es una suite de oficina en linea cuyos servicios son gratuitos de utilizarse. Recientemente, Zoho ha abierto su API (Application Programming Interface) permitiendo que terceras aplicaciones como IUSSU, puedan utilizar los servicios de Zoho.

---

9 Anexo A

10 Anexo B

11 Anexo C

12 Anexo D

13 Anexo E

14 Zoho. Zoho.Work.Online [En línea] http://www.zoho.com 2007 [citado el 05-07-07]

El reproductor en línea de Google mail, es uno de los servicios gratuitos que ofrece Google en su idioma inglés disponible para todos sus usuarios.

El reproductor xsfp-player[15] es Open Source Software distribuido bajo una licencia BSD y puede ser utilizado y modificado por cualquiera, inclusive con propósitos comerciales.

15 ZUARDI Fabricio. XSPF Music Web Player. [En linea] http://musicplayer.sourceforge.net 2005 [citado el 05 de Julio de 2007]

3. METODOLOGÍA DE DESARROLLO

En Ingeniería de Software y Gerencia de Proyectos, una metodología es un conjunto de prácticas que pueden seguirse y repetirse para producir software. Usualmente una metodología toca muchas disciplinas incluido el análisis, la especificación, el diseño, la codificación, las pruebas y el tipo de documentación.

Entre las metodologías en Ingeniería de Software están la Programación Estructurada, La Programación Orientada a Objetos y El Desarrollo de Software Ágil entre otros.

La Metodología del presente proyecto es el Desarrollo de Software Ágil[16]. Ésta metodología cuenta con varios métodos de desarrollo, algunos más antiguos que otros como (Scrum, Extreme Programming, EVO, UP) que en términos generales siguen los cuatro postulados del Manifiesto por el Desarrollo Ágil de Software[17] que son:

- Individuos e interacciones sobre procesos y herramientas
- Software que funciona sobre documentación exhaustiva
- Colaboración con el cliente sobre negociación de contratos
- Responder ante el cambio sobre seguimiento de un plan.

Los métodos del Desarrollo de Software Ágil son opuestos a la separación de fases del diseño de software tradicional en el que expertos diseñan el código sin escribir una sola línea de código y programadores ejecutan el diseño tratando de no alterarlo en lo más mínimo.

16 Página Oficial The Agile Alliance [http://www.agilealliance.com] Mas información sobre

En la práctica, ningún diseño puede considerarse correcto hasta que ha sido codificado y probado, ya que siempre están los elementos sorpresa, como una especificación de hardware que no funciona como se previó, una restricción de alguna librería que no estaba documentada o cualquier otro tipo de imprevisto, que muy posiblemente terminarán repercutiendo en el diseño. En el Desarrollo de Software Ágil, se intentan minimizar éstos riesgo de desarrollo de software, reduciendo al mínimo la distancia entre el diseño y la codificación. Un desarrollo se planea en pequeños ciclos llamados iteraciones que típicamente tienen una duración de una a cuatro semanas y que constituyen un proyecto en miniatura con los pasos necesarios para ir desarrollando la aplicación de forma incremental. Este desarrollo incremental del proyecto implica que los requerimientos, planes y demás estimaciones también evolucionan y son refinados durante el curso de las iteraciones.

No se intentan especificar todas las características antes de comenzar a codificar, sino que se comienza con una especificación básica y de allí se crea inmediatamente un primer prototipo funcional.

Las prácticas de un método ágil a otro pueden variar. En la Tabla No 2 pueden apreciarse las prácticas más comunes de cada una de las metodologías ágiles más conocidas. En el presente Proyecto *IUSSU* se utilizará una metodología mixta en la que se desarrollará una combinación de prácticas de Scrum y XP acordadas por los integrantes de este proyecto (las que aparecen sombreadas en la Tabla No 2)

Desarrollo Ágil de Software en español en Agile Spain [http://agile-spain.com]

17 Manifesto for Agile Software Development [http://www.agilemanifesto.org]

Práctica	Descripción
SCRUM	
Pre-Juego planeación y estacionamiento	Durante el Pre-Juego, todos contribuyen a crear una lista de características, casos de uso, mejoras, defectos y similares grabados en el la Reserva del Producto. Un propietario del producto es designado como su poseedor y las peticiones son mediadas por medio de este individuo. Durante esta sesión se determina suficiente trabajo para la primera iteración, aunque es usual que sea mucho mayor.
El Planeamiento del Sprint	Antes de cada iteración -o Sprint – se hacen dos reuniones consecutivas. Primero los apostadores se reunen para refinar y repriorizar la Reserva del Producto elegir las metas para la siguiente iteración usualmente dirigidas por la que mayor riesgo de negocio represente. Y en la segunda reunión los miembors del equipo y el Propietario del producto se reunen para considerar como van a lograr los requerimientos.
Sprint	El trabajo es usualmente organizado en iteraciones de 30 días calendario y cada una es llamada Sprint
Equipos auto dirigidos y auto organizados	Durante una iteración el director del proyecto no guía al equipo en el cómo va a realizar las metas de la iteración. Para solucionar este problema el equipo es embestido con la autoridad y los recursos para encontrar su propio camino y solucionar sus problemas.
Reunión Scrum	Cada dia de trabajo a la misma hora y en el mismo lugar se realiza un meeting con los miembros del equipo en un círculo en el cual las preguntas Scrum son hechas a cada miembro del equipo. 1. Qué se ha hecho desde la ultima reunión Scrum? 2. Que se hará entre esta y la siguiente reunión Scrum? 3. Qué esta bloqueando la realización de las metas de la iteración? 4. Alguna tarea que sea necesario añadir ? 5. Ha aprendido o decidido algo nuevo de relevancia para otros miembros del proyecto?
No agregar a una iteración	Durante una iteración el director del proyecto no puede agregar trabajo al equipo o a los individuos interrumpiendo el foco que es llevado. En muy raros casos si si algo debe ser agregado, algo deberá ser removido entonces.
Director Scrum cortafuegos	El director Scrum se asegura de que el trabajo del equipo no sea interrumpido por causas externas. Si estas ocurren las remueve y lidia con todos las políticas externas necesarias.
Decisiones en una hora	El valor de tomar una mala decisión es superior al de no tomar ninguna decisión es promovido.

Práctica	Descripción
Bloqueos retirados en un dia	Los bloqueos reportados en la reunión Scrum son idealmente removidos antes de la siguiente reunión
Pollos y Cerdos	Aunque pueden asistir observadores a las reuniones, solamente los miembros del equipo Scrum deben hablar en estas reuniones.
Equipos de siete	Scrum puede escalar a grandes proyectos pero se recomienda que cada equipo no tenga un número superior a 7 miembros, para macroproyectos se recomienda crear multi-equipos.
Preferencia por un salón común	Idealmente el equipo trabaja junto en un salón de proyectos común en lugar de cubículos separados. Aunque espacio privado todavía debe estar disponible para otras actividades.
Construcción diaria	Al menos se debe verificar el código del proyecto una vez al día
Revisión Sprint	Al final de cada iteración existe una reunión de revisión de una duración máxima de 4 horas dirigida por el director Scrum, el equipo, el propietario del producto. Hay una prueba del producto. El propósito incluye informar a los apostadores de las funciones del sistema, su diseño, sus fortalezas, debilidades, el esfuerzo del equipo y los futuros focos de problemas. Retroalimentación y lluvia de ideas son animadas para la futura dirección pero no se realizan nuevos compromisos durante la reunión. Luego en la siguiente reunion de planeación serán cuando se hagan.
XP	
Cliente a tiempo completo con el equipo	Unos o más clientes se sientan más-o-menos a tiempo completo con el equipo; se espera que sean expertos del tema y que estén autorizados para tomar decisiones con respecto a los requisitos de la aplicación y su prioridad. La contribución del cliente incluye la explicación detallada a los programadores de las características resumidas brevemente en tarjetas de la historia, la participación, la clarificación, e incluso la planeación de las pruebas de aceptación en colaboración con un programador. La participación activa del cliente es de vital importancia en los proyectos acertados.
Pequeños y frecuentes lanzamientos	Lanzamiento evolutivo. No aplicable para todos los proyectos.
Pruebas: Pruebas de aceptación y pruebas del cliente	Las prácticas de pruebas son muy importantes en XP.

Práctica	Descripción
	Todas las características deben tener test funcionales de aceptación.
Pruebas atutomáticas	Unidades de pruebas deben ser escritas para la mayor parte del código
El Juego de Lanzar el planteamiento	El propósito de este juego es determinar le siguiente lanzamiento operacional con el máximo valor desde el punto de vista del cliente para el software. Típicamente en medio día de un día de sesión, los clientes escriben tarjetas que describen las características y los desarrolladores estiman la complejidad y el tiempo requerido. Los clientes entonces eligen cual será el siguiente lanzamiento bien sea por 1) colocando una fecha y colocando cartas hasta que se estime el total de tiempo disponible o 2) eligiendo cartas y calculando el lanzamiento en el estimado de las fechas.
Juego de Planeación de iteraciones	El propósito es elegir qué historias implementar y planear y localizar las tareas para la iteración. Los clientes eligen las tarjetas para la iteración. Por cada una los programadores crean una lista de tareas en las tarjetas o en un tablero para realizar las historias. Entonces se estima la longitud de las tareas si alguna tarea no puede ser estimada tiene que ser replanteada.
Diseño simple	Evitar diseño especulativo para futuros cambios. Evitar crear componentes generalizados que no se requieran inmediatamente. Evitar duplicación de código y tener relativamente el mínimo conjunto de clases y métodos que puedan comprenderse fácilmente
Programación en parejas	Toda la producción de código es creada por dos programadores en un computador, rotándose el uso de los dispositivos de entrada. Las parejas pueden cambiar frecuentemente para diferentes tareas. El observador está haciendo revisión del código en tiempo real y tal vez pensando con mas lucidez que quien digita, considerando pruebas y cosas por el estilo Ciertamente la productividad de un equipo no está determinado por el número de manos que digitan. XP asegura que de esta forma, se disminuye el tiempo de revisión del código.
Frecuente Replanteamiento	Replantear en el contexto de XP es el esfuerzo continuo de simplificar cada vez en un grano más fino el código y elementos más grandes del diseño, al tiempo que todavía se asegura que todas las pruebas pasan. Es decir, limpiando el código y el diseño, sin que la funcionalidad

Práctica	Descripción
	cambie.
El código es del equipo	Cualquier par de programadores puede mejorar cualquier código. El valor de es 'su' código y por lo tanto es 'su' problema no es válido. Si la posibilidad de mejorar algún punto es vislumbrada, es responsabilidad del que lo notó. Es responsabilidad del equipo todo el código, no solamente la parte que les fue asignada.
Integración continua	Todo código verificado es continuamente reintegrado y probado en una máquina determinada para tal fin en un proceso automatizado que está compilando y corriendo todas las unidades de pruebas.
Ritmo continuo	Frecuentemente el trabajar tiempo-extra es correctamente un síntoma de problemas graves en el trabajo, que conlleva a programadores malhumorados, con menos creatividad y con problemas familiares. XP no recomienda trabajar tiempo-extra.
Estándares de codificación	Todos necesitan seguir el mismo estilo de codificación.
EVO	
Definir el top 10 de requerimientos clave	Evo recomienda la definición de los requerimientos críticos. En cada iteración deben ser revisados y redefinidos.
Definir funciones especificas	La funciones Evo describen lo que el sistema hace. Evo no promueve un detallado y extenso análisis de requerimientos pero si requiere una clara definición de las funciones esperadas para la siguiente iteración.
Definir las especificaciones de desempeño	Evo promueve describir el desempeño del sistema. Qué tan bien trabaja el sistema, sus beneficios y cómo estos afectan el ambiente. Los atributos de desempeño están ligados a las funciones y se clasifican en tres categorías. 1) Calidad 2)Capacidad 3)Cantidad de Recursos que utilizan
Definir de forma clara y en lo posible de forma cuantificada las especificaciones	Evo promueve un balance entre muy poco y demasiado detalle en los requerimientos.
Utilizar Planguage para las especificaciones	Planguaje es un lenguaje estructurado de Evo para las especificaciones tanto de requerimientos como de diseño, su uso es opcional, pero se recomienda.
Entregas evolutivas	Evo hace énfasis en la entrega de soluciones parciales a producción de forma temprana, para así obtener una temprana retroalimentación de lo que es valioso en el negocio y así orientar las siguientes entregas
Medir el impacto de las soluciones entregadas	Esta información es importante para dirigir la evolución del proyecto iteración por iteración.
Definir especificaciones de diseño	Las ideas de diseño son guardadas en Planguage e incrementalmente desarrolladas.

Práctica	Descripción
	Las especificaciones de diseño deberían explicar por qué y hasta qué grado cumplen con los requerimientos.
Especificar pruebas y medidas de los requerimientos	No dejar requerimientos como buen desempeño sino cuantificarlos y diseñar pruebas que verifiquen dichas cuantificaciones.

Tabla 2: Prácticas y Descripción de Metodologías Ágiles.

4. ANÁLISIS Y DISEÑO

La parte más difícil de construir software es precisamente decidir qué construir.

Frederick Brooks.

En informática, el análisis es el momento en el que se toman las decisiones para desarrollar el proyecto después de considerar los límites, características y posibles soluciones para lograr satisfacer los requerimientos funcionales y no funcionales de la aplicación que se desea implementar. Y en la fase del diseño, se comienza a establecer el proceso de configuración mental, lo que está por venir para llevar el proyecto a la obra. Es la búsqueda de una solución o un conjunto de las mismas.

4.1 DEFINICIÓN DEL SISTEMA PROPUESTO

Como recomiendan la mayoría de líderes de modelado ágil, establecer y reiterar una visión común es importante. Aunque pareciese una idea 'obvia', no hay dos personas que tengan exactamente la misma visión y propósito de sus proyectos.

En el libro de Agile and Iterative Development : A Manager's Guide de Craig Larman recomienda un ejercicio propuesto originalmente por Geoffrey Moore, en la que se sigue el formato de la Tabla No 3 para establecer por medio de aserciones una visión mas concreta y unificada.

Para	(el mercado al que se dirige)
Quien	(frase de necesidad u oportunidad)
El	(nombre del producto)
Es un	(categoría del producto)
Que	(beneficio clave, razón para comprarlo)
A diferencia de	(primera alternativa de la competencia)
Nuestro producto	(frase de primera diferencia que ofrece el producto)

Tabla 3: Formato para Establecer la Visión del Proyecto

Siguiendo dicho ejercicio, las respuestas de los dos integrantes del proyecto fueron:

- Opinión Integrante 1: "*Para* cualquier tipo de usuario mayor de 13 años, *quien* no posea el software o el medio físico de cargar la información (archivos), *el* Organizador *IUSSU es un* Administrador de Colecciones,que *a diferencia de* los reproductores on-line, *nuestro producto* es libre posee la capacidad de almacenar los archivos del usuario y reproducirlos o visualizarlos"

- Opinión Integrante 2: "*Para* jóvenes mayores de 13 años *quienes* cuenten con una conexión a internet y tengan colecciones de música, libros o imágenes que quieran organizar. *El* Organizador *IUSSU es un* Administrador de Colecciones, que *a diferencia de* las aplicaciones de escritorio que administran colecciones, *nuestro producto* funciona on-line"

4.2 DESCRIPCIÓN DETALLADA DEL SISTEMA PROPUESTO

La pregunta es, ¿qué debe comenzarse a hacer en la primera Iteración y en las

siguientes?. Las metodologías de ágil desarrollo, promueven tomar esta decisión por cada iteración y tomar como base para determinar ésta decisión la evaluación de dos factores.

El primero el riesgo o impacto que tendría implementar un requerimiento determinado, el riesgo incluye características como la complejidad, el esfuerzo, la inexperiencia, la incertidumbre o la duración para realizarlo.

El segundo factor, la prioridad para el cliente. Esto implica que las elecciones de las siguientes características a ser implementadas en la siguiente iteración son determinadas por el cliente y lo que es valioso para él, ya que después de todo es él quien conoce el mercado al que desea acceder.

En el Proyecto *IUSSU* se consideraron ambos factores, para determinar los próximos requerimientos a ser implementados.

Siguiendo una de las prácticas recomendadas por Scrum, se realizó una sesión de Planeación o Pre-Juego, por cada una de las tres iteraciones que se realizaron en las que cada vez se creó una lista de características deseables para la aplicación, se enumeraron algunos casos de uso e incluso se señalaron posibles mejoras de la aplicación para el futuro.

Con la última versión de éste listado se implementó una tabla que puede apreciarse en la Tabla No. 4 llamada la Reserva del Producto. Allí puede apreciarse cada uno de los requerimientos planteados al inicio de cada una de las iteraciones, en gris oscuro los de la primera iteración, en gris medio los de la segunda iteración y en gris más claro, los de la tercera iteración. Adicionalmente se intentó clasificar la categoría del requerimiento, la prioridad, la duración y el riesgo de implementar dicho requerimiento.

La prioridad es un valor cualitativo determinado por el cliente, aunque dada la naturaleza del proyecto en la que el cliente y los desarrolladores comparten la misma entidad, la prioridad representa en este caso el valor que tiene para los desarrolladores de *IUSSU* ver implementado ese requerimiento. Para intentar darle un valor cuantitativo, se tomó una escala del 1 al 5, siendo 5 la máxima prioridad y 1 la menor.

La duración está tomada en semanas y es un valor estimado que estuvo variando en cada una de las iteraciones.

El riesgo es el impacto que tendría la implementación de dicho requerimiento estimado por los desarrolladores. Siendo éste otro valor cualitativo, se intentó cuantificarlo tomando una escala del 1 al 5, 5 como el máximo riesgo y 1 como el menor.

Requerimiento	Categoría	Prioridad	Duración Estimada	Riesgo de Implement.
Reproducir Música	Característica	5	3 semanas	5
Reproducir Vídeo	Mejora	2	6 semanas	5
Visualizar Documentos	Característica	5	2 semanas	3
Visualizar Imágenes	Característica	5	2 semanas	3
Restringir el espacio de almacenamiento	Mejora	3	4 semanas	4.5
Análisis de métodos de pago por Internet	Mejora	3.5	1 semana	3

Escenario de Registro de Usuario en la aplicación	Caso de Uso	5	1.5 semanas	4
Investigación sobre Términos y Condiciones Legales de reproducción y almacenamiento en Internet	Derechos de Autor	4.5	1.5 semanas	1
Escenario de Agregar nuevo Item a Colección	Caso de Uso	5	3 semanas	5
Escenario de Buscar Item dentro de Colecciones Públicas o de la Colección del Usuario	Caso de Uso	4	3 semanas	4
Escenario de Crear Nuevo Tipo de Colección	Caso de Uso	4	3 semanas	5
Revisión del rating del material de los usuarios.	Característica	4	2 semanas	3
Escenario de Modificación de Características de la Interfaz del Usuario	Caso de Uso	3.5	4 semanas	3.5
Escenario de Eliminar Item de Colección	Caso de Uso	4	2 semanas	4
Investigación Servidores compatibles con Ruby on Rails	Tecnología	5	8 semanas	5
Escenario de Modificación de Item de Colección	Caso de Uso	4.5	2 semanas	4
Documentación			6 meses ++	
Implementación de Pruebas Automáticas en Ruby	Tecnología	4.5	Duración del	4.5

			Proyecto	
Documentación RDoc	Tecnología	4.5	Duración del Proyecto	4.5

Tabla 4: Reserva del Producto.

4.3 DESCRIPCIÓN DETALLADA DE CADA PROCESO

Para definir las primeras especificaciones de la aplicación, se implementaron casos de uso de alto nivel y un esbozo del flujo entre las páginas web.

4.3.1 Casos de Uso del Proyecto IUSSU

Se identificaron dos diferentes roles o actores: el usuario y el administrador del sistema. El usuario quien utiliza la aplicación para subir la información de sus colecciones y visualizarla. Y el administrador utiliza la aplicación para gestionar las cuentas de los usuarios, verificar el tipo de información que los usuarios tienen, así como la cantidad de espacio de la que disponen según el tipo de cuenta.

- CASO DE USO DE REGISTRO

1. El usuario solicita registrarse, llenando un formulario en el que suministra la información de sus datos personales. Y aceptando los términos y condiciones de uso del servicio.
2. El sistema envía un correo de verificación a la dirección suministrada por el usuario. Solicitándole al usuario la confirmación de activación de la cuenta.
3. El sistema crea una cuenta con las característica por defecto.
4. El usuario puede acceder a su cuenta.

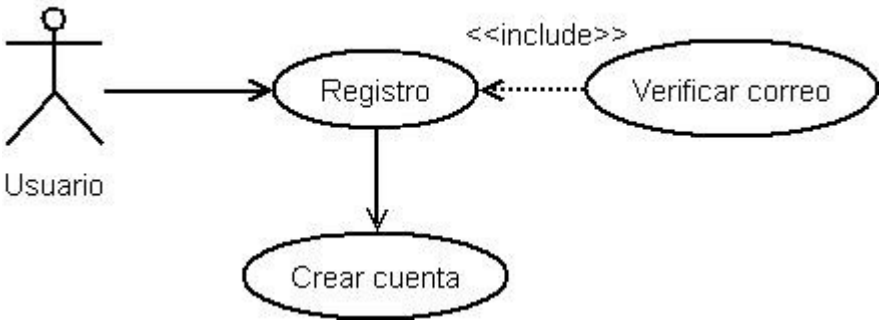


Diagrama 3: Caso de Uso Registro

- CASO DE USO LOGIN

1. El usuario se ha registrado previamente y suministra sus datos de nick y contraseña.
2. El sistema despliega la página de inicio de la cuenta correspondiente al usuario.

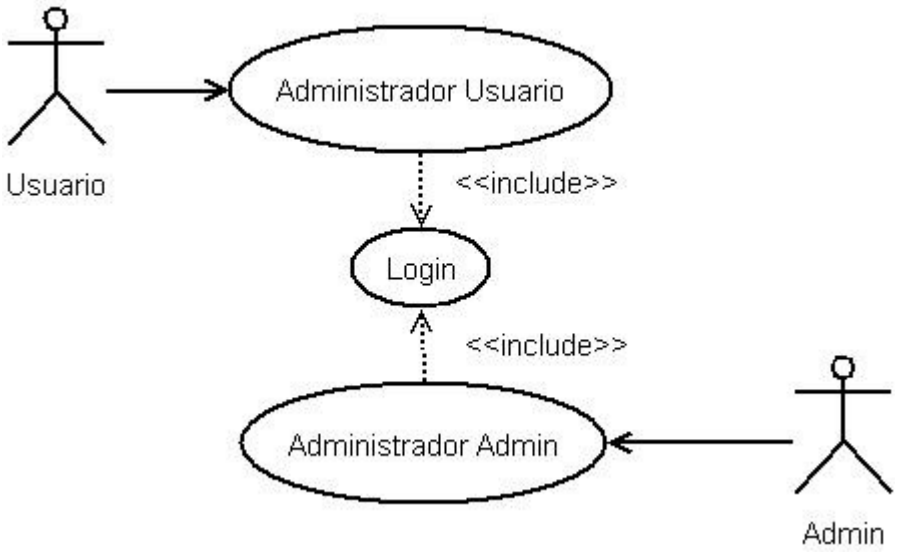


Diagrama 4: Caso de Uso Login

- CASO DE USO AGREGAR NUEVA COLECCIÓN

1. El usuario solicita crear una nueva colección.
2. El sistema le pregunta si desea utilizar un tipo de colección ya existente o si desea crear un nuevo tipo de colección.
3. a. El usuario selecciona un tipo de colección existente y elige cuales de los campos de dicha colección deberán ser obligatorios y cuales deberán ser públicos.
4. b. El usuario selecciona un nuevo tipo de colección. Selecciona el número y tipo de campos que deberá tener dicha colección. Posteriormente seleccionará cuáles de estos campos serán obligatorios y cuales podrán desplegarse públicamente.
5. El usuario suministra los nuevos datos de la colección a ser creada y cuando lo decida da por terminada la creación de la nueva colección.
6. El sistema visualizará la nueva colección dentro del conjunto de colecciones del usuario.

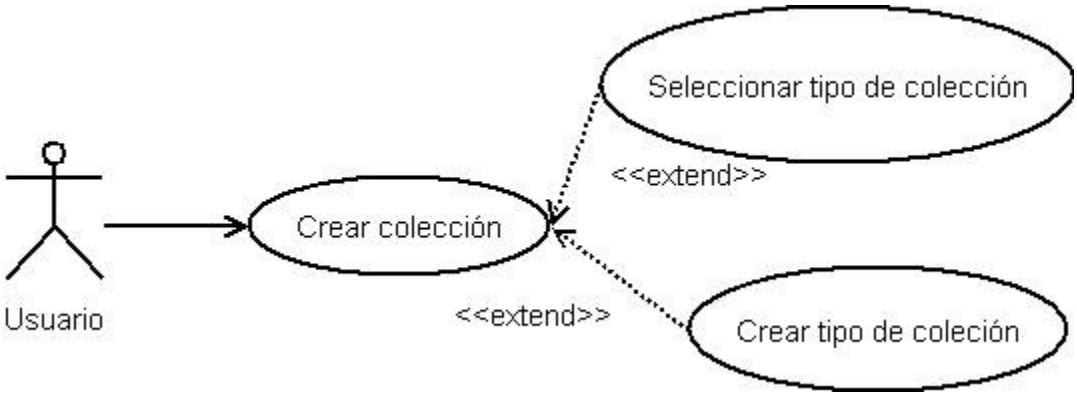


Diagrama 5: Caso de Uso Agregar Nueva Colección

- CASO DE USO AGREGAR ITEM A COLECCIÓN

1. El usuario selecciona la colección a la que desea agregar el nuevo item.

2. Dependiendo del tipo de colección, el sistema exigirá que se le suministren los valores para los campos de dicha colección.
3. Cuando el usuario da por terminada la creación del nuevo item.
4. El sistema procesa la información y si no encuentra ningún problema mostrará la colección actualizada con el nuevo item.

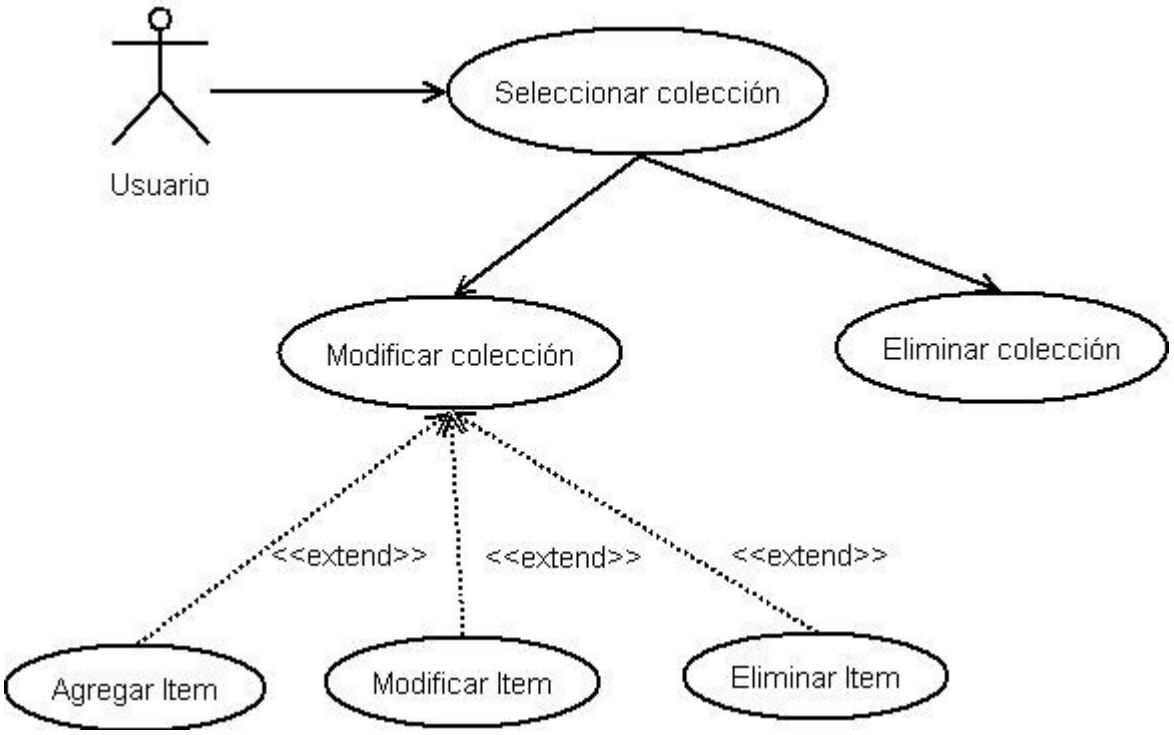


Diagrama 6: Caso de Uso Agregar Item a Colección

- CASO DE USO REPRODUCCIÓN DE ARCHIVO DE MÚSICA

1. El usuario selecciona un archivo de música de los disponibles en su colección o sube un archivo de música a su colección.
2. El sistema añade el elemento a la cola de reproducción en curso. Si no hay elementos en la cola de reproducción, el archivo comenzará a reproducirse. Las

opciones de adelantar, retroceder, pausar, detener y variar el volumen de la reproducción en curso estarán disponibles en todo momento para el usuario.

(Ver Diagrama No. 7)

- CASO DE USO VISUALIZACIÓN DE ARCHIVO DE TEXTO

1. El usuario selecciona un archivo de texto de los disponibles en su colección o sube un archivo de texto a su colección.
2. El sistema le permite visualizar un pdf con la información del archivo de texto seleccionado.

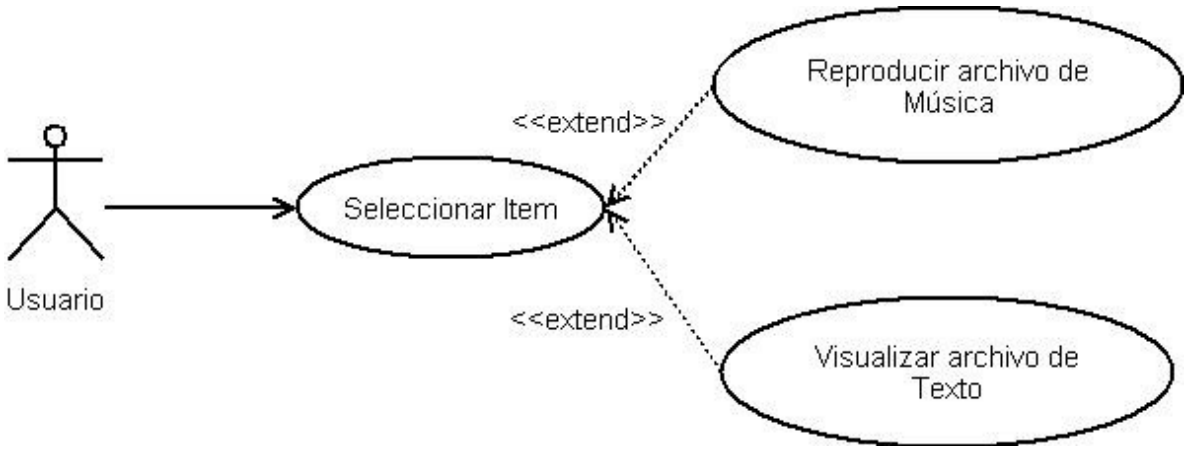


Diagrama 7: Caso de Uso Reproducir/Visualizar Item

- CASO DE USO ADMINISTRACIÓN DE USUARIOS

1. El administrador del sistema ingresa al sistema suministrando su nick y su contraseña. Estos datos fueron creados con la aplicación y ningún usuario puede registrarse como administrador.
2. El sistema le ofrece al administrador un listado con los usuarios registrados y las principales características de sus cuentas.

3. El administrador podrá seleccionar, crear, modificar o eliminar a los usuarios y/o el contenido de sus colecciones.
4. El sistema actualizará las solicitudes del administrador.

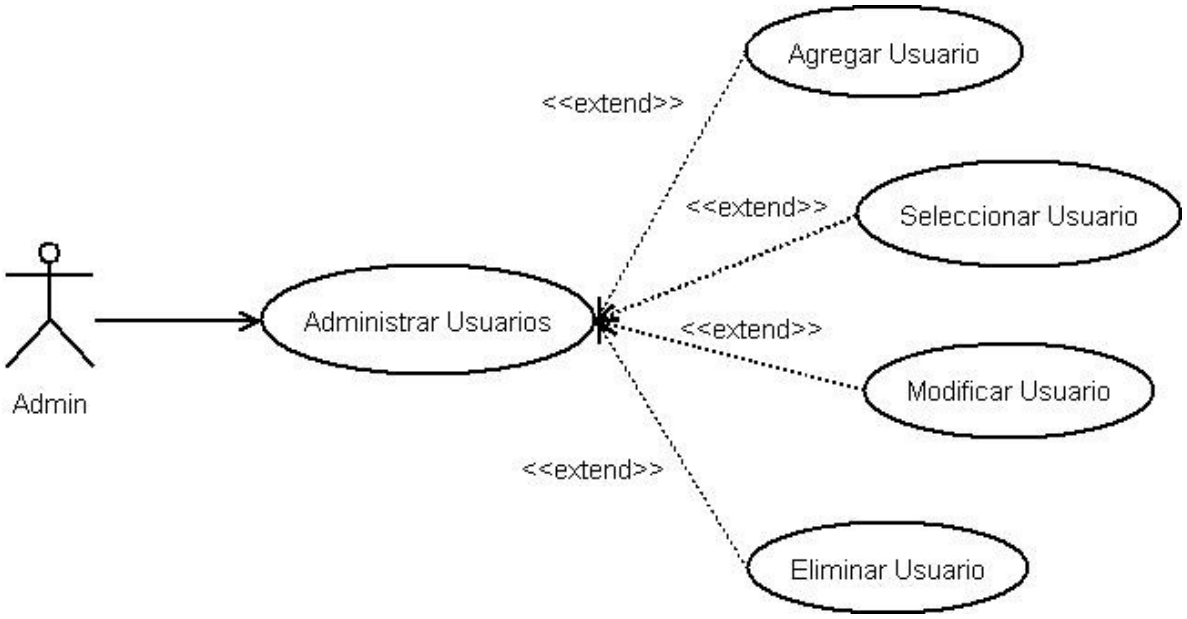


Diagrama 8: Caso de Uso Administración de Usuarios

4.3.2 Flujo de Páginas Web (Diseño del Sistema Propuesto)

Para visualizar el diseño de la interfaz con la que el usuario y el administrador del sistema interactuarán, en el Diagrama No 5 puede apreciarse un dibujo del flujo de páginas web del usuario y en el Diagrama No 6 uno más detallado del mismo flujo.

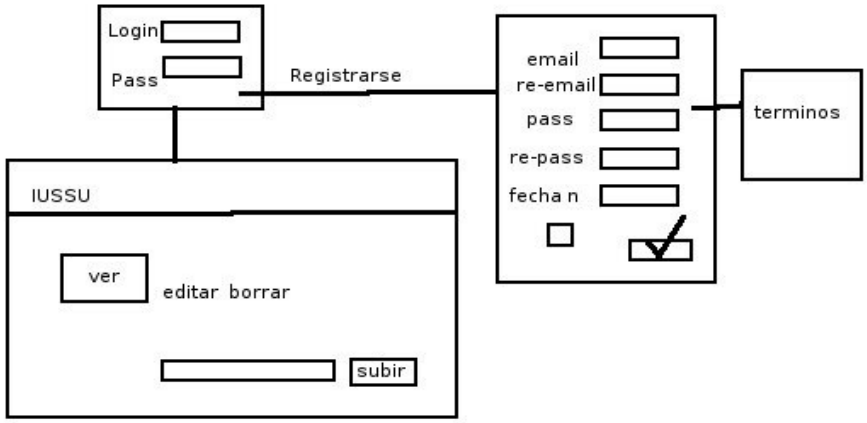


Figura 5: Flujo de Páginas Web del Usuario

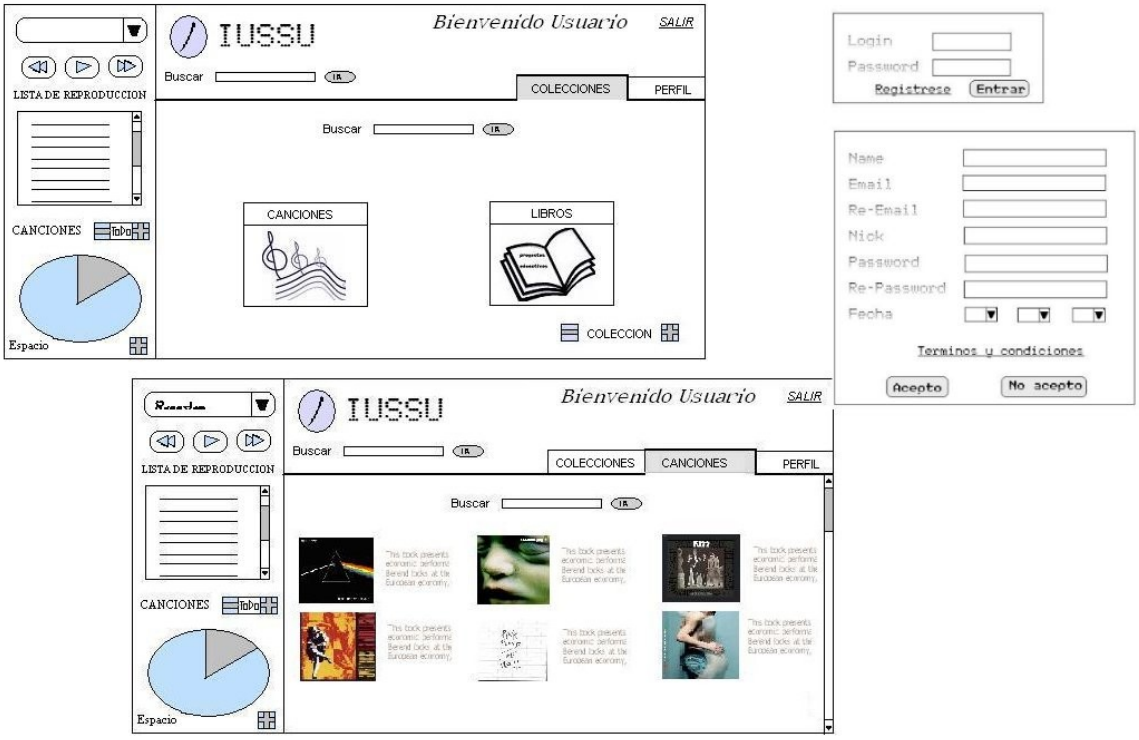


Figura 6: Flujo de Páginas Web del Usuario Detallado

4.4 DICCIONARIO DE DATOS

Es común que al diseñar un sistema de información se deba utilizar unos términos estándar. En el Proyecto *IUSSU* se tendrán en cuenta los siguientes estándares:

- Se utilizarán los tipos de datos que maneja por defecto el framework *Rails* .

Nombre del tipo de Datos	Descripción
:primary_key	Llave primaria
:string	Cadena de caracteres
:text	Cadena de caracteres extendida
:integer	Entero
:float	Flotante
:timestamp	Fecha calculada en milisegundos transcurridos desde las 00:00:00 del 1 enero de 1970
:time	Hora
:date	Fecha
:binary	Binario
:boolean	Booleano
:datetime	Fecha y Hora

Tabla 5: Tipos de Datos en Rails

- Los nombres de los campos tendrán menos de quince dígitos, los tres primeros referentes a la tabla a la cual pertenecen, comenzando en minúscula y el resto de dígitos para la descripción del campo. Utilizando guiones para separar palabras compuestas.

La base de datos trabaja con las siguientes tablas:

- TABLA DE USUARIOS

Nombre de la tabla: usuarios

Descripción de la tabla: Tabla en la que se almacenan los datos de cada uno de los usuarios del sistema.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla usuarios
usu_nombre	:string	Nombre del usuario
usu_email	:string	Email del usuario
usu_password	:string	Password del usuario
usu_verificado	:boolean	Campo para verificar si el usuario tiene acceso al correo que registró.
usu_nacimiento	:datetime	Fecha de nacimiento del usuario

Tabla 6: Tabla de Usuarios

- TABLA DE COLECCIONES

Nombre de la tabla: coleccions

Descripción de la tabla: Tabla en la que se relacionan los diferentes tipos de colecciones que existen por usuario.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla coleccions
usuario_id	:integer	Referencia a la llave primaria de la tabla usuarios
col_nombre	:string	Nombre de la colección

col_imagen	:string	Imagen que distingue la colección
col_descri	:text	Descripción de la colección
col_tipo	:string	Uno de los cuatro tipos posibles de colecciones.
col_reproductor	:string	Referencia al reproductor de musica si la colección es de tipo MP3
col_lista	:string	Referencia a la lista de reproducción si la colección es de tipo MP3

Tabla 7: Tabla de Colecciones

- TABLA DE CANCIONES

Nombre de la tabla: musiks

Descripción de la tabla: Tabla en las que se almacena la información de la colección de canciones del usuario.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla musiks
coleccion_id	:integer	Referencia a la tabla de coleccions
mus_titulo	:string	Titulo de la canción.
mus_autor	:string	Intérprete o autor de la canción.
mus_archivo	:string	Referencia al archivo mp3 de la canción.
mus_comentario	:text	Comentario de la canción

mus_imagen	:string	Referencia a la imagen que representa la canción
mus_propiet	:string	Nick del usuario que posee la canción

Tabla 8: Tabla de Canciones

- TABLA DE DOCUMENTOS

Nombre de la tabla: documentos

Descripción de la tabla: Tabla en las que se almacena la información de la colección de documentos del usuario.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla documentos
coleccion_id	:integer	Referencia a la tabla coleccions
doc_titulo	:string	Título del documento
doc_autor	:string	Autor del documento
doc_archivo	:string	Referencia al archivo del documento
doc_comentario	:text	Comentario del documento
doc_imagen	:string	Referencia a la imagen que representa al documento
doc_propiet	:string	Nick del usuario que posee el documento

Tabla 9: Tabla de Documentos

- TABLA DE IMAGENES

Nombre de la tabla: imagens

Descripción de la tabla: Tabla en las que se almacena la información de la colección de imagenes del usuario.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla imagens
coleccion_id	:integer	Referencia a la tabla coleccions
ima_titulo	:string	Título de la imagen
ima_autor	:string	Autor de la imagen
ima_archivo	:string	Referencia al archivo de la imagen
ima_comentario	:text	Comentario de la imagen
ima_dimensiones	:string	Dimensiones del archivo de la imagen
ima_propiet	:string	Nick del usuario que posee el documento

Tabla 10: Tabla de Imágenes

- TABLA DE OTROS

Nombre de la tabla: otros

Descripción de la tabla: Tabla en las que se almacena la información de la colección de variados archivos del usuario.

NOMBRE DE LOS CAMPOS	TIPO DE DATO	DESCRIPCION
id	:primary_key	Llave primaria de la tabla otros
coleccion_id	:integer	Referencia a la tabla coleccions
otr_titulo	:string	Título del archivo
otr_autor	:string	Autor del archivo
otr_archivo	:string	Referencia al archivo
otr_comentario	:text	Comentario del archivo
otr_imagen	:string	Referencia a la imagen que representa al archivo
otr_propiet	:string	Nick del usuario que posee el archivo

Tabla 11: Tabla de Otros

4.5 MODELO ENTIDAD RELACIÓN

A continuación una variación del tradicional modelo entidad relación, para representar de una forma gráfica los atributos y las relaciones entre las tablas de la base de datos de la aplicación IUSSU.

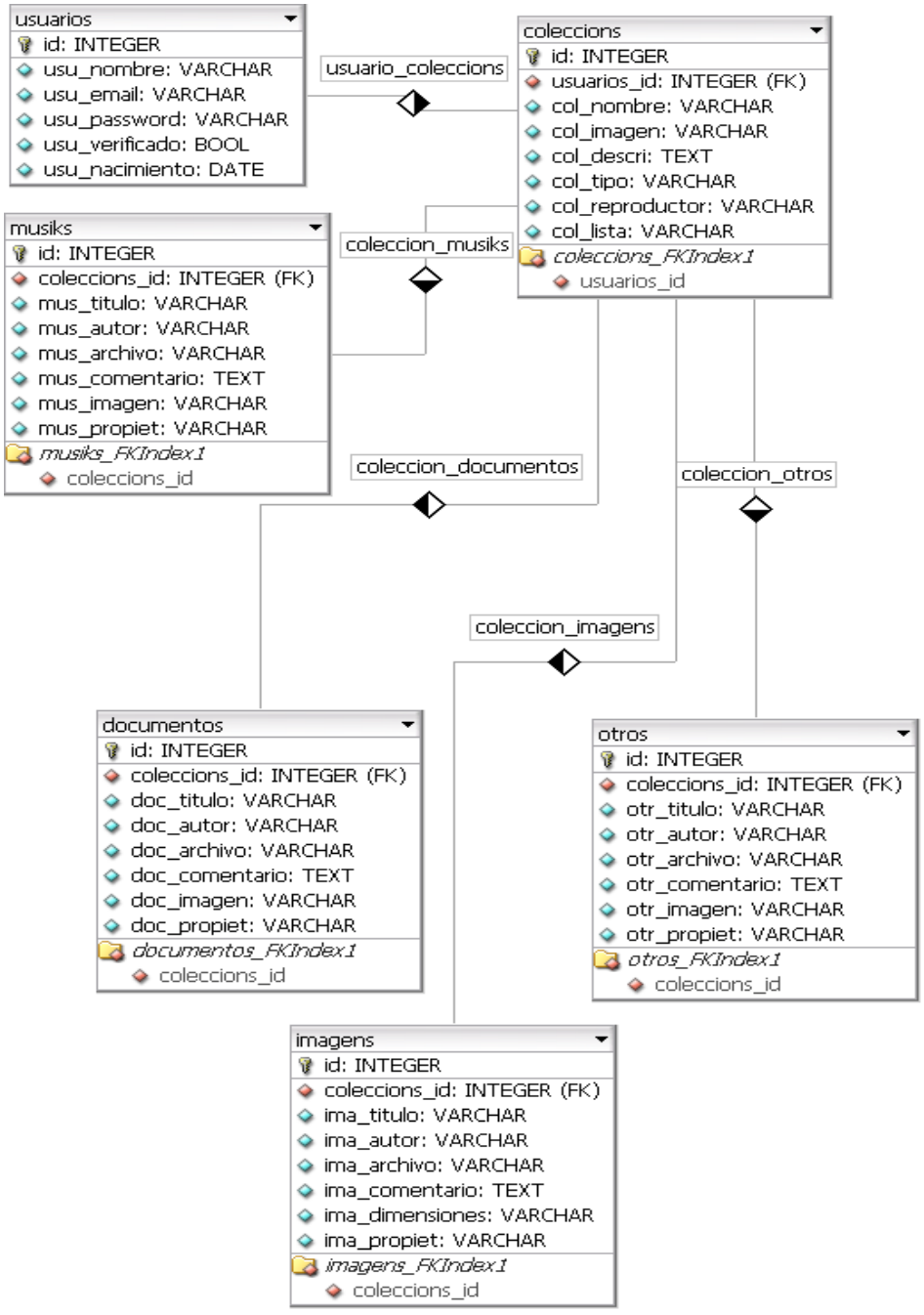


Diagrama 9: Modelo Entidad-Relación

5. DESARROLLO

Para obtener un producto final, se ha de reunir un mínimo de requerimientos para que tanto el proceso de creación de software como su posterior ejecución sean posibles.

5.1 ESPECIFICACIONES TÉCNICAS

Se dividen en dos partes, las especificaciones técnicas del hardware y las especificaciones técnicas del software; a su vez cada una de éstas se divide en las especificaciones para el servidor y las especificaciones para los clientes.

5.1.1 Software

Desde el punto de vista del servidor, se utilizaron las siguientes características:

- Sistema Operativo Fedora Core 6 con las herramientas para desarrollo y para funcionar como servidor.
- *Ruby* 1.8.5 con librerias de desarrollo, irb, rdoc.
- RubyGems 0.9.2
- *Ruby On Rails* 1.2.3
- Lighttpd 1.4.13 con soporte para Fastcgi
- MySQL 5.0.27
- ImageMagick 6.2.5.5 y RMagick 1.15.6

- Editor de textos SciTE 1.73
- DNS dinámico con Dyndns

Desde el punto de vista de la aplicación cliente, se requiere para el funcionamiento

- Un navegador web preferiblemente Firefox 2.0.0.4 o en su defecto IE 7.0 con plugins de flash player 8.0 o superior y adobeReader.

5.1.2 Hardware

Desde el punto de vista del servidor se utiliza:

- Un PC con procesador AMD64 +3000, board KN3 Neo, 1.5 GB RAM, disco SATA 80 GB.
- Router Ethernet D-Link DI-604
- Servicio de Internet Banda Ancha 700 MB con TVCable

Desde el punto de vista del cliente necesita:

- Computador con los requerimientos básicos para un navegador web Firefox o IE 7.0.
- Servicio de Internet.

6. GLOSARIO[18]

FRAMEWORK es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

RUBY es un lenguaje de programación reflexivo y orientado a objetos creado por el programador japonés Yukihiro Matsumoto en 1993. Combina una sintaxis inspirada en Python, Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

RUBY ON RAILS también conocido como RoR o simplemente Rails es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración. El lenguaje de programación Ruby permite la metaprogramación, de la cual Rails hace uso, lo que resulta en una sintaxis que muchos de sus usuarios encuentran muy legible. Rails se distribuye a través de RubyGems que es el formato oficial de paquete y canal de distribución de librerías y aplicaciones Ruby.

IUSSU palabra en latín que significa 'Por orden de'.

18 Definiciones tomadas de la Wikipedia en español, artículos en línea en [http://es.wikipedia.org/]

LICENCIAS OPEN SOURCE las licencias deben cumplir diez condiciones para ser consideradas licencias de software abierto:

1. Libre redistribución: el software debe poder ser regalado o vendido libremente.
2. Código fuente: el código fuente debe estar incluido u obtenerse libremente.
3. Trabajos derivados: la redistribución de modificaciones debe estar permitida.
4. Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
5. Sin discriminación de personas o grupos: nadie puede dejarse fuera.
6. Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
7. Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa
8. La licencia no debe ser específica de un producto: el programa no puede licenciarse sólo como parte de una distribución mayor.
9. La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
10. La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

SOFTWARE LIBRE[19] para que una aplicación sea considerada software libre debe cumplir con cuatro condiciones:

1 La libertad de usar el programa, con cualquier propósito (libertad 0).
2 La libertad de estudiar cómo funciona el programa, y adaptarlo a las necesidades del usuario (libertad 1).
3 La libertad de distribuir copias (libertad 2).

19 Condiciones tomadas de la Página Oficial de la Free Software Foundation [http://gnu.org]

4 La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3).

MVC MODELO VISTA CONTROLADOR es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el controlador es el Sistema de Gestión de Base de Datos y el modelo es el modelo de datos.

MODELO en el patrón MVC es la representación específica de la información con la cual el sistema opera. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos; por ejemplo, no permitiendo comprar un número de unidades negativo, calculando si hoy es el cumpleaños del usuario o los totales, impuestos o portes en un carrito de la compra.

VISTA en el parón MVC éste presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

CONTROLADOR en el patrón MVC éste responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

LICENCIA MIT es una de tantas licencias de software que ha empleado el MIT (Massachusetts Institute of Technology) a lo largo de su historia, y quizás debería llamarse más correctamente licencia X11, ya que es la licencia que llevaba este Software originario del MIT en los años 80. Pero ya sea como MIT o X11, su forma es idéntica. El texto de la licencia no tiene copyright, lo que permite su modificación. No obstante esto puede no ser recomendable e, incluso, muchas

voces dentro del Open Source lo desaconsejan, a no ser que se indique que es una modificación, y no la versión original.

LICENCIA BSD es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Pertenece al grupo de licencias de software Libre. Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

CREATIVE COMMONS está inspirada en la licencia GPL (General Public License) de la Free Software Foundation (Sin embargo no es un tipo de licenciamiento de software libre). La idea principal es posibilitar un modelo legal y ayudado de herramientas informáticas para así facilitar la distribución y el uso de contenidos para el dominio público. Ofrece una serie de licencias, cada una con diferentes configuraciones o principios como el derecho del autor original a dar libertad para citar su obra, reproducirla, crear obras derivadas, ofrecerlo públicamente y con diferentes restricciones como no permitir el uso comercial o respetar la autoría original.

7. CONCLUSIONES

Desde el inicio del desarrollo del proyecto IUSSU se estableció que su objetivo era el de establecer las ventajas y desventajas que el desarrollo de una aplicación escrita en el lenguaje de programación *Ruby* y el framework *Ruby on Rails* podría representar para un desarrollador que desee crear una aplicación web.

Así, tras la realización de éste proyecto, concluimos que entre las ventajas de éste tipo de desarrollo están:

- Su rápida curva de aprendizaje. Los programadores del proyecto IUSSU se enfrentaron al reto de escribir una aplicación en un lenguaje por completo desconocido para ellos y tras un relativo corto tiempo de familiarización con la sintaxis de *Ruby* (que incluye características como la innecesaria declaración de variables, un verdadero recolector de basura, que trabaja con todos los objetos de Ruby y que evita que el programador deba mantener una relación de referencias, el usualmente innecesario uso de signos de puntuación como puntos y comas y paréntesis para separar sentencias) pudieron verificar la rapidez con la que estuvieron en capacidad de construir sus propias sentencias lógicas en el lenguaje.
- La rapidez de desarrollo. Gracias a características del framework *Ruby on Rails* como las 'migraciones'[20] y los 'scaffoldings'[21] se puede tener una aplicación funcional en prácticamente cinco minutos, así como la seguridad de que probar cualquier cambio sobre ésta primera aplicación base tampoco tomará mayor

20 Característica de RoR que permite modificar el esquema de la base de datos usando lenguaje

tiempo. Los programadores de IUSSU se sintieron motivados por la presencia de dichas características y no dudan de su importancia directa en el desarrollo de la aplicación, aunque en el caso de los 'scaffoldings' hayan sido reemplazados por completo.

- Gran cantidad de información en diversos formatos como foros, listas de correos, tutoriales, videos y libros abundan en internet e hicieron que conseguir fuentes para ésta investigación fuese una tarea fácil.
- Compatibilidad, flexibilidad y extensiones. *Ruby* cuenta con numerosas librerías y extensiones como JRuby que es un intérprete del lenguaje Java en *Ruby* que permite utilizar librerías escritas en Java como Itext (una librería gratuita para la manipulación de documentos PDF) y que en algún punto del desarrollo se consideró en implementar en IUSSU. *Ruby on Rails* cuenta con facilidades para desarrollar e instalar plugins que son modificaciones o extensiones para realizar tareas concretas y repetitivas como la subida de archivos, el sistema de tags o la validación de una fecha entre otros. Los desarrolladores de IUSSU utilizaron cuatro de estos plugins, acts_as_taggable_on_steroids, file_column, smtp_plugin y validates_date_time
- Seguridad. El trabajar con una aplicación base de código abierto que sigue estándares de arquitectura, ampliamente utilizada por diversas personas alrededor del mundo garantiza que cuando una vulnerabilidad de seguridad es descubierta, su anuncio es un hecho público, así como su solución.
- Facilidad de escalabilidad. Cuando se piensa en desarrollar una aplicación web, la escalabilidad es importante. Con unos cuantos cientos de usuarios un sistema que presta servicios en internet tendrá que expandirse indiscutiblemente hacia múltiples servidores e inclusive múltiples bases de datos. Aunque éste punto se escapa del carácter de IUSSU en éstos momentos. Es importante saber que *Ruby on Rails* considera y permite éste tipo de Arquitecturas distribuidas.

---

de programación Ruby, de forma que las instrucciones para modificar la base de datos se

Entre las desventajas se encontró que

- La documentación se encuentra en el idioma inglés y en japonés si bien ése no fue un problema para los programadores de IUSSU, quienes leen con fluidez el inglés, sí pueden preveer ésto como una desventaja para otros programadores de habla hispana, porque aunque existe información en español, no se compara con la que se encuentra en el idioma inglés.

- El Hosting de una aplicación *Rails* requiere que la compañía que ofrece el servicio de hosting tenga las últimas versiones de *Ruby* y de *Ruby on Rails* instaladas y ciertamente la mayoría de las compañías de hosting están acostumbradas a operar con PHP /Perl y no cuentan con la experiencia y/o con el software, lo que hace que las compañías que sí prestan este servicio no lo ofrezcan por un precio económico. Para bordear éste inconveniente los programadores de IUSSU decidieron dejar la aplicación en el mismo servidor que utilizaron para desarrollo al menos en ésta fase del proyecto.

- Y por último, lo que los desarrolladores de IUSSU denominaron el peligro de la caja negra. El peligro que siempre puede representar para el conocimiento el tener un algo que funciona con facilidad pero que no se entiende en su totalidad y al que imparablemente tienden todas las soluciones tecnológicas de éste tiempo.

---

mantienen centralizadas e independientes del tipo de base de datos que se vaya a utilizar.

21 Característica de RoR que permite generar de forma automática vistas por defecto para un determinado modelo de datos.

BIBLIOGRAFÍA

BECK, Kent, CUNNINGHAM, Ward, THOMAS, Dave et al. Manifesto for Agile Software Development. 2001 [citado el 27 de Septiembre de 2006] Disponible en Internet <URL http://www.agilemanifesto.org/>.

BURD, Barry. Ruby On Rails For Dummies [ebook]. Wiley Publishing, Inc., Indianapolis, Indiana. 2007. p.349. ISBN 978-0-470-08120-4

ECO, Umberto. Cómo elaborar una tesis. Barcelona. GEDISA S.A. 1991

FOUCAULT, Michel. Las palabras y las cosas. Barcelona. Siglo XXI (October 2002)

FOWLER, Chad. Rails Recipes [ebook]. Versión 2006-5-15. The Pragmatic Bookshelf, 2006. p.297. ISBN 0-9776166-0-6.

HEINEMEIER HANSSON, David. Rails Web development that doesn't hurt [en línea]. 37 Signals. [citado en 27 de septiembre de 2006] Página Oficial. Disponible en Internet <URL: http://www.rubyonrails.org/>.

INSTITUTO COLOMBIANO DE NORMAS TECNICAS. Compendio – Tesis y Otros Trabajos de Grado. Santafé de Bogotá D. C., ICONTEC, 2006. NTC 1486, NTC 1075, NTC 1487, NTC 1160, NTC 1308, NTC 1307, NTC 4490. ISBN 958-9383-07-6

Licencia Open Source [en línea]. Wikipedia La Enciclopedia Libre. Actualizado el 2

de marzo de 2006. [citado en 27 de septiembre de 2006] Texto en español. Disponible en Internet: <URL: http://es.wikipedia.org/wiki/Definici%C3%B3n_de_Open_Source>.

MANSFIELD, Richard. CSS For Dummies [ebook]. Wiley Publishing, Inc., Indianapolis, Indiana. 2005. p385. ISBN 0-7645-8425-1

MARSHALL, Kevin. Web Services On Rails [ebook]. O'Reilly. 2006. p.32. ISBN 0-596-52796-9

Publicaciones y Ediciones Spes, S.A. Diccionario Abreviado Latino-Español Español-Latino. Tercera Edición. Barcelona: Gráficas Marina, 1950. p.310.

PALACIO, Juan. Los Cuatro Principios del Manifiesto Ágil [en línea]. Navegapolis.net 1 de Junio de 2006. [citado el 27 de Septiembre de 2006] Artículo disponible en Internet <URL http://www.navegapolis.net/contet/view/355/83/>

Ruby [en línea]. Wikipedia La Encyclopedia Libre. Actualizado el 7 de septiembre de 2006. [citado en 27 de septiembre de 2006] Texto en español. Disponible en Internet: <URL:http://es.wikipedia.org/wiki/Ruby>.

Ruby (programming language) [en línea]. Wikipedia The Free Encyclopedia. Actualizado el 27 de septiembre de 2006. [citado en 28 de septiembre de 2006] Texto en inglés. Disponible en Internet: <URL:http://en.wikipedia.org/wiki/Ruby_programming_language>.

Ruby on Rails [en línea]. Wikipedia La Encyclopedia Libre. Actualizado el 10 de septiembre de 2006. [citado en 27 de septiembre de 2006] Texto

en español. Disponible en Internet: <URL:http://es.wikipedia.org/wiki/Ruby_on_Rails>.

Ruby on Rails (programming language) [en línea]. Wikipedia The Free Encyclopedia. Actualizado el 28 de septiembre de 2006. [citado en 28 de septiembre de 2006] Texto en inglés. Disponible en Internet: <URL:http://en.wikipedia.org/wiki/Ruby_on_Rails>.

Ruby community. Ruby A Programmer's Best Friend [en línea]. Radiant CMS. [citado en 27 de septiembre de 2006] Página Oficial en Inglés. Disponible en Internet <URL: http://www.ruby-lang.org/en/>.

RUSTAD, Aaron. Ruby on Rails and J2EE: Is there room for both? [en línea]. Ibm 12 de Julio de 2005 [citado en 27 de septiembre de 2006] DeveloperWorks. Disponible en Internet <URL: http://www-128.ibm.com/developerworks/web/library/wa-rubyonrails/#figure3>.

SCHMITT, Christopher y Otros. Professional Css – Cascading Style For Web Design [ebook]. Wiley Publishing, Inc., Indianapolis, Indiana. 2005. p.459. ISBN-13: 978-0-7645-8833-4

THOMAS, Dave y HUNT, Andy. Programming in Ruby [ebook]. Addison-Wesley, 13 de septiembre de 2000. p.14.

THOMAS, Dave. Agile Web Development with Rails [ebook]. Second Edition. The Pragmatic Bookshelf, 2006. p.637. ISBN 0-9776166-3-0.

ANEXO A
GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee,

you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute

a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms,

do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable.

However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of

this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR

LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

ANEXO B

RUBY LICENSE

Ruby is copyrighted free software by Yukihiro Matsumoto <matz@netlab.co.jp>.
You can redistribute it and/or modify it under either the terms of the GPL
(see COPYING.txt file), or the conditions below:

1. You may make and give away verbatim copies of the source form of the
   software without restriction, provided that you duplicate all of the
   original copyright notices and associated disclaimers.
2. You may modify your copy of the software in any way, provided that
   you do at least ONE of the following:
   a) place your modifications in the Public Domain or otherwise
      make them Freely Available, such as by posting said
      modifications to Usenet or an equivalent medium, or by allowing
      the author to include your modifications in the software.
   b) use the modified software only within your corporation or
      organization.
   c) rename any non-standard executables so the names do not conflict
      with standard executables, which must also be provided.
   d) make other distribution arrangements with the author.
3. You may distribute the software in object code or executable
   form, provided that you do at least ONE of the following:
   a) distribute the executables and library files of the software,
      together with instructions (in the manual page or equivalent)
      on where to get the original distribution.
   b) accompany the distribution with the machine-readable source of
      the software.

c) give non-standard executables non-standard names, with instructions on where to get the original software distribution.

d) make other distribution arrangements with the author.

4. You may modify and include the part of the software into any other software (possibly commercial). But some files in the distribution are not written by the author, so that they are not under this terms. They are gc.c(partly), utils.c(partly), regex.[ch], st.[ch] and some files under the ./missing directory. See each file for the copying condition.
5. The scripts and library files supplied as input to or produced as output from the software do not automatically fall under the copyright of the software, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this software.
6. THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

ANEXO C

THE MIT LICENSE

Copyright (c) <2004> <37Signals>
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

ANEXO D

BSD LICENSE LIGHTTPD

Copyright (c) 2004, Jan Kneschke, incremental
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the 'incremental' nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR

CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

ANEXO E

CREATIVE COMMONS ATTRIBUTION 2.5

Attribution 2.5

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Collective Work" means a work, such as a periodical issue, anthology or

encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

b. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

c. "Licensor" means the individual or entity that offers the Work under the terms of this License.

d. "Original Author" means the individual or entity who created the Work.

e. "Work" means the copyrightable work of authorship offered under the terms of this License.

f. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor

hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;

b. to create and reproduce Derivative Works;

c. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;

d. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

e. For the avoidance of doubt, where the work is a musical composition:

    i. Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

    ii. Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

f. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast)

of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions.The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

b. If you distribute, publicly display, publicly perform, or publicly digitally

perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

b. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, neither party will use the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time.

Creative Commons may be contacted at http://creativecommons.org/.

ANEXO F

MANUAL DE USUARIO

INTRODUCCIÓN

Gracias por preferir los servicios de *IUSSU*.

IUSSU es un administrador de colecciones que permite almacenar todos sus archivos de forma gratuita.

REGISTRO EN IUSSU

Por favor siga los pasos a continuación:

1. Diríjase al link ¿Registrarse?, que se encuentra ubicado en la parte inferior del identificador, como lo señala la imagen a continuación

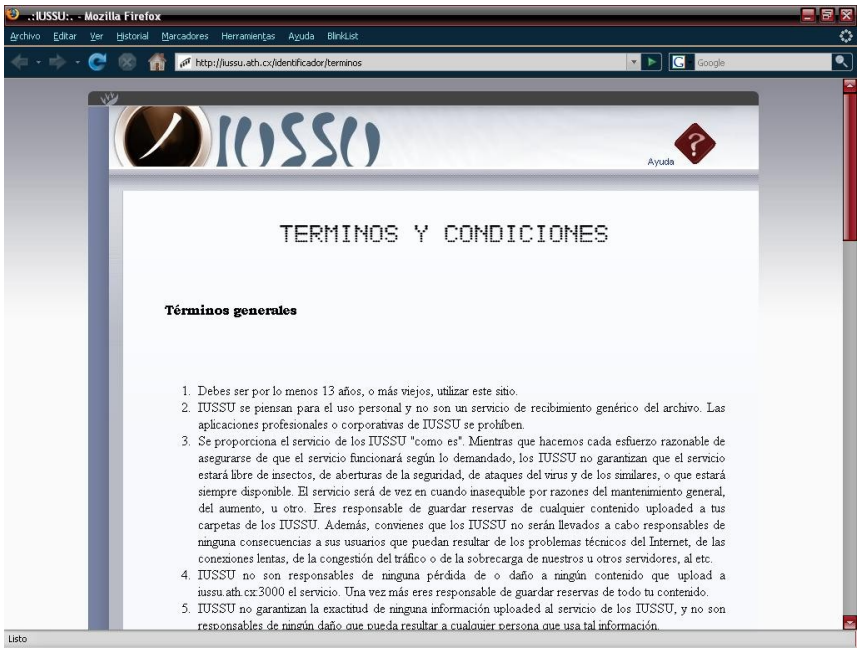
2. Haga clic en ¿Registrarse?

3 Ingrese sus datos en el formulario de registro, El formulario es como la imagen a continuación

Nota: Es importante que el email concuerde con el Re-Email, así como el password sea igual al de Re-Password, de lo contrario aparecerán mensajes de error.

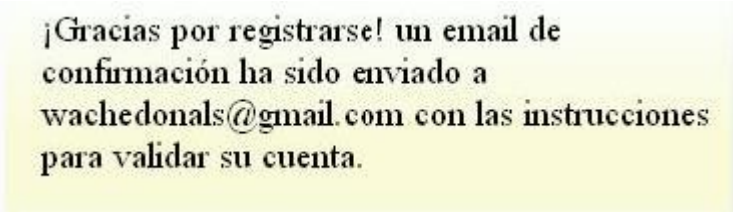
El correo es para que pueda activar su cuenta en *IUSSU*.
No es necesario que ingrese el password de su correo, si no que digite el que usted desee y debe ser mayor a 6 caracteres.

4. Ingrese su fecha de nacimiento, si usted es menor de 13 años es decir si nació después del 1 enero1994 no podrá acceder a IUSSU.
5. Lea atentamente los términos y condiciones a los que puede acceder con darle clic sobre el link que se encuentra en la parte inferior del formulario. Si desea continuar, con darle clic sobre el cuadro aceptara los Términos y condiciones de *IUSSU*

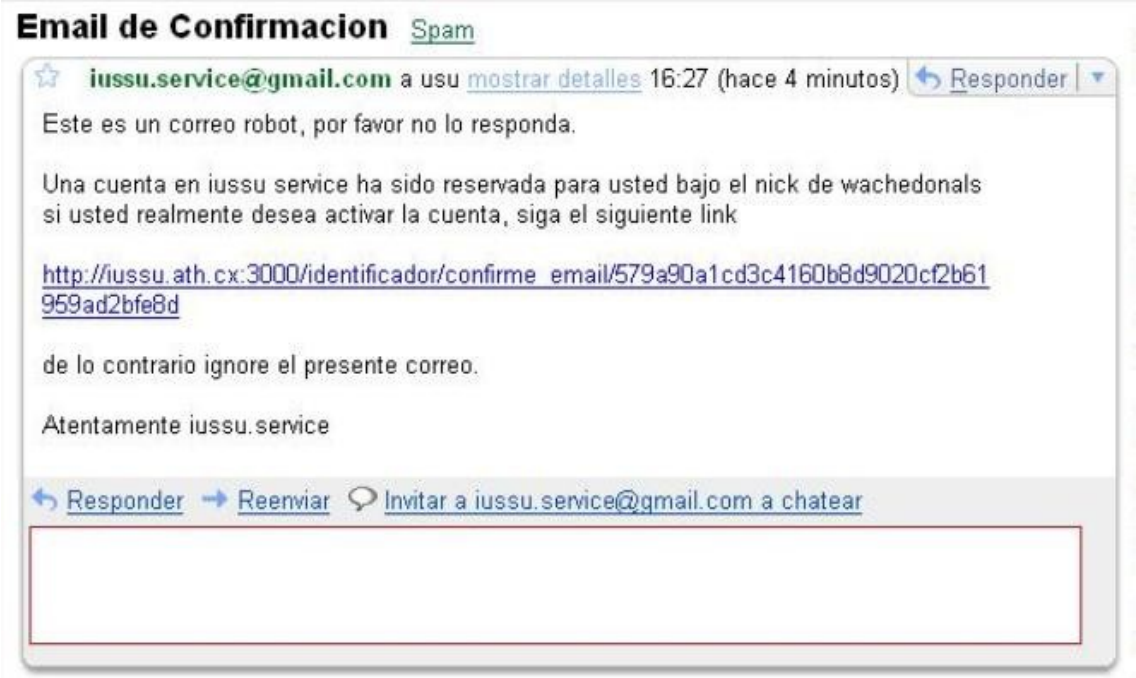

.::IUSSU:. - Mozilla Firefox
Archivo Editar Ver Historial Marcadores Herramientas Ayuda BlinkList
http://iussu.ath.cx/identificador/terminos
IUSSU
Ayuda
TERMINOS Y CONDICIONES
Términos generales
1. Debes ser por lo menos 13 años, o más viejos, utilizar este sitio.
2. IUSSU se piensan para el uso personal y no son un servicio de recibimiento genérico del archivo. Las aplicaciones profesionales o corporativas de IUSSU se prohiben.
3. Se proporciona el servicio de los IUSSU "como es". Mientras que hacemos cada esfuerzo razonable de asegurarse de que el servicio funcionará según lo demandado, los IUSSU no garantizan que el servicio estará libre de insectos, de aberturas de la seguridad, de ataques del virus y de los similares, o que estará siempre disponible. El servicio será de vez en cuando inasequible por razones del mantenimiento general, del aumento, u otro. Eres responsable de guardar reservas de cualquier contenido uploaded a tus carpetas de los IUSSU. Además, convienes que los IUSSU no serán llevados a cabo responsables de ninguna consecuencias a sus usuarios que puedan resultar de los problemas técnicos del Internet, de las conexiones lentas, de la congestión del tráfico o de la sobrecarga de nuestros u otros servidores, al etc.
4. IUSSU no son responsables de ninguna pérdida de o daño a ningún contenido que upload a iussu.ath.cx:3000 el servicio. Una vez más eres responsable de guardar reservas de todo tu contenido.
5. IUSSU no garantizan la exactitud de ninguna información uploaded al servicio de los IUSSU, y no son responsables de ningún daño que pueda resultar a cualquier persona que usa tal información.
Listo

6. Si usted acepto los Términos y condiciones, y además ya ingreso el resto de la información en los campos del formulario de registro, haga clic sobre el botón de *registrarse*.

7. Aparece un mensaje de agradecimiento, por favor verifique que el correo es el suyo. Por ejemplo nuestro usuario Wachedonals


¡Gracias por registrarse! un email de confirmación ha sido enviado a wachedonals@gmail.com con las instrucciones para validar su cuenta.

8. Ingrese a su correo y revise que tenga en la bandeja de entrada un mensaje de *IUSSU*, colóquese sobre el link y haga clic, este link activara la cuenta.
Nota: En el servidor de Yahoo el correo de activación ingresa al spam.


Email de Confirmacion
Spam
iussu.service@gmail.com a usu mostrar detalles 16:27 (hace 4 minutos)
Responder
Este es un correo robot, por favor no lo responda.
Una cuenta en iussu service ha sido reservada para usted bajo el nick de wachedonals
si usted realmente desea activar la cuenta, siga el siguiente link
http://iussu.ath.cx:3000/identificador/confirme_email/579a90a1cd3c4160b8d9020cf2b61959ad2bfe8d
de lo contrario ignore el presente correo.
Atentamente iussu.service
Responder
Reenviar
Invitar a iussu.service@gmail.com a chatear

9. Ingrese su nick y la contraseña del registro en el Identificador, y haga clic sobre el botón *login*.

COMO CREAR UNA COLECCION EN IUSSU.

En IUSSU usted puede crear unas cuatro tipos de colecciones.

Documentos Imagen MP3 Otro

Continuemos con los pasos

10. Seleccione el tipo de colección que desea crear, y después haga clic en el botón Crear Nueva colección.

La colección puede ser editada en su presentación y físicamente ser a su gusto.

11. Haga clic sobre el link editar que se encuentra al lado derecho de la colección.

12. Edita los campos que desee.

13. Si desea cambiar la imagen haga clic en examinar, aparecerá una ventana emergente, por favor seleccione la imagen que identificara a su colección.

Nota: Sólo podrá seleccionar imágenes.

13. Ingrese la descripción de su colección.

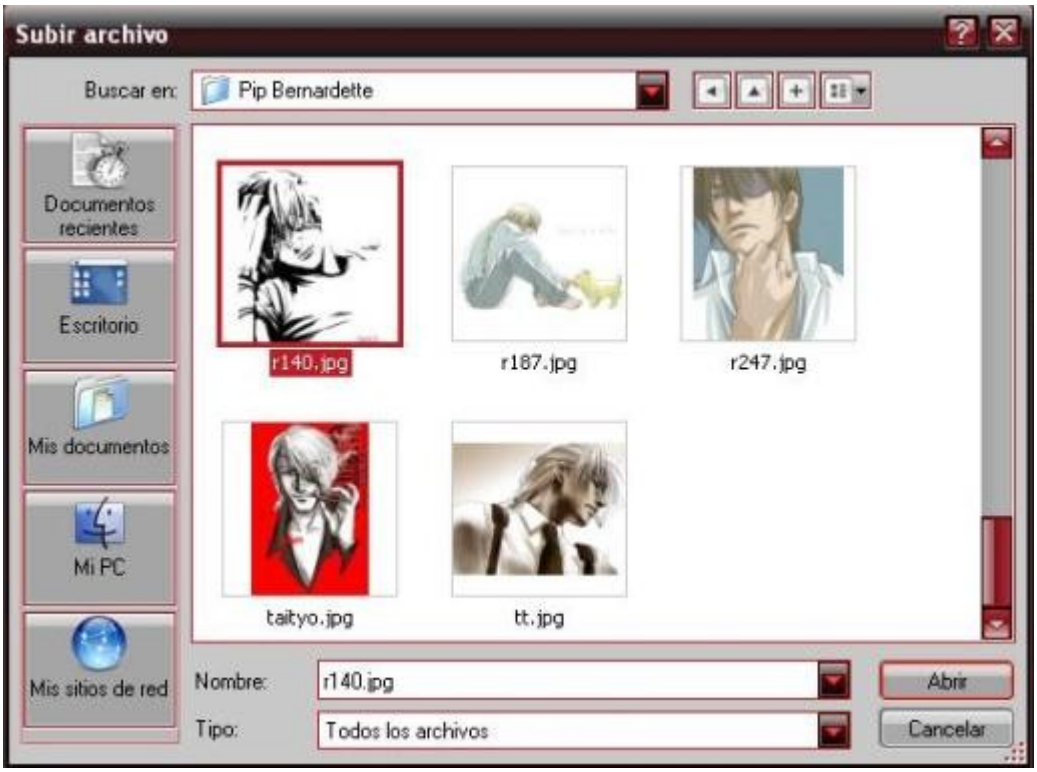
14. Haga clic sobre el botón crear.

COMO SUBIR ARCHIVOS EN LA COLECCIÓN

IUSSU permite que agregue a su colección archivos según el tipo de colección, para ingresar continuemos con los pasos

15. Coloque el cursor sobre la imagen de la colección y haga clic sobre la imagen así podrá ingresar a la colección creada.

16. Como puede ver aun no tiene ningún archivo así que haga clic sobre el botón de examinar, aparecerá una ventana emergente como la imagen a continuación, seleccione el archivo que desea agregar.



(En la imagen podemos ver el ejemplo con la colección de imágenes)

17. Haga clic sobre el botón de subir y espere unos segundos.

18. Haga esto cuantas veces usted desee.

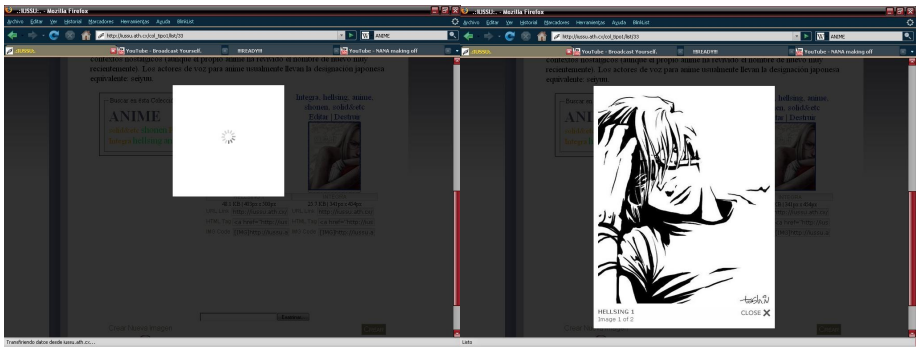
ARCHIVOS

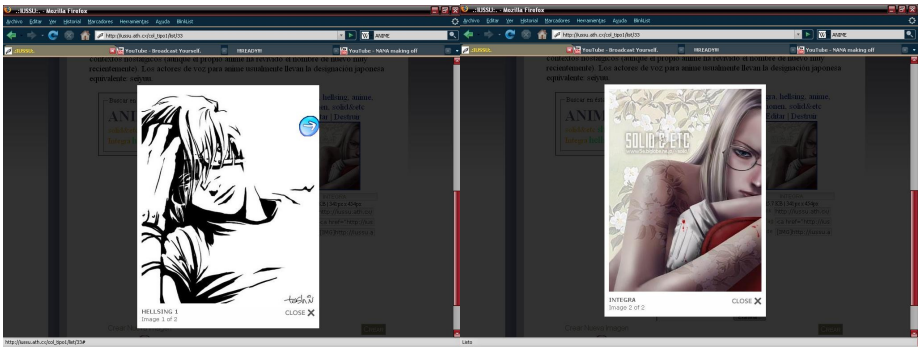
Los archivos son editables en su nombre y descripción, en algunos casos en la imagen que los distingue y también pueden ser destruidos. Además pueden descargarse y visualizarse.

Para entender mejor continuemos.

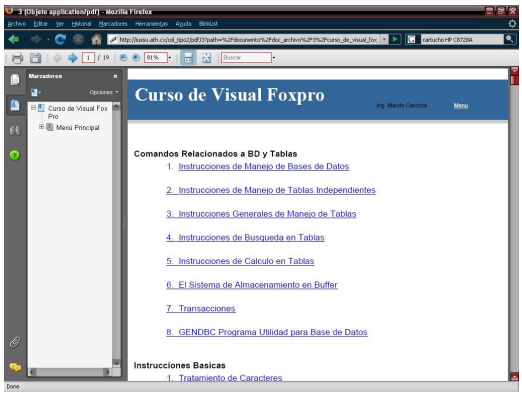
19. Los archivos que usted agrego a las colecciones pueden modificarse según el tipo de colección es decir:

- En la colección Imágenes usted puede modificar el nombre, los tags, la imagen o destruirlo. Para visualizar las imágenes, solo tiene que dar clic sobre la imagen y podrá verla sobre la colección y si da clic sobre la flecha ya sea hacia delante o atrás podrá ver la imagen anterior o siguiente como lo vemos a continuación, en la parte inferior podrá cerrar la visualización

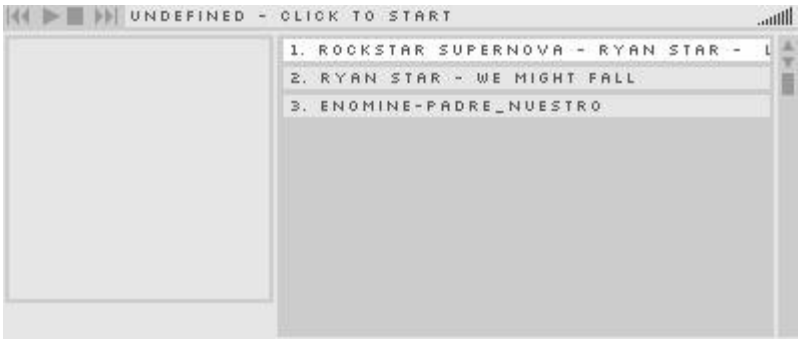


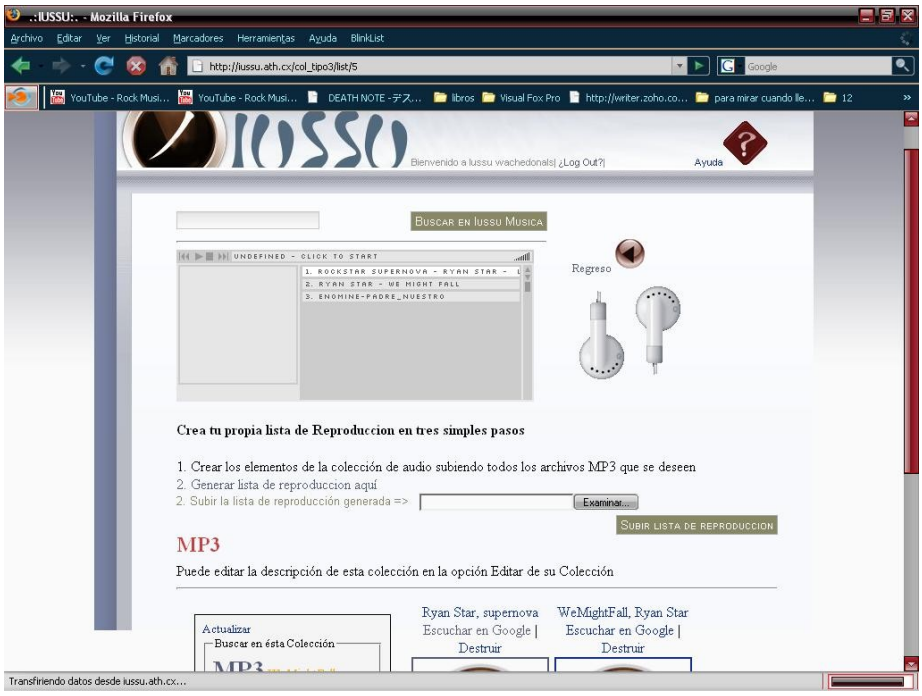


- En la Colección Documentos usted puede modificar el nombre, la imagen, los tags. Para visualizarlos, haga clic sobre el botón Ver Documento (según el formato del archivo).


Curso de Visual Foxpro
Comandos Relacionados a BD y Tablas
1. Instrucciones de Manejo de Bases de Datos
2. Instrucciones de Manejo de Tablas Independientes
3. Instrucciones Generales de Manejo de Tablas
4. Instrucciones de Busqueda en Tablas
5. Instrucciones de Calculo en Tablas
6. El Sistema de Almacenamiento en Buffer
7. Transacciones
8. GENDBC Programa Utilidad para Base de Datos
Instrucciones Basicas
1. Tratamiento de Caracteres

- En la Colección MP3 usted puede modificar el nombre, los tags y la imagen. Así como puede modificar la lista de reproducción que usted descarga. Para reproducir los archivos mp3 solo debe seguir los pasos, subir la canción y descargar la lista luego subirla y el reproductor iniciara.(si lo desea puede reproducir en el reproductor de Google)




.:IUSSU:. - Mozilla Firefox
Archivo Editar Ver Historial Marcadores Herramientas Ayuda BlinkList
http://iussu.ath.cx/col_tipo3/list/5
Bienvenido a Iussu wachedonals| ¿Log Out?|
Ayuda
Buscar en Iussu Musica
UNDEFINED - CLICK TO START
1. ROCKSTAR SUPERNOVA - RYAN STAR -
2. RYAN STAR - WE MIGHT FALL
3. ENGMINE-PADRE_NUESTRO
Regreso
Crea tu propia lista de Reproduccion en tres simples pasos
1. Crear los elementos de la colección de audio subiendo todos los archivos MP3 que se deseen
2. Generar lista de reproduccion aquí
2. Subir la lista de reproducción generada =>
Examinar...
Subir lista de reproduccion
MP3
Puede editar la descripción de esta colección en la opción Editar de su Colección
Actualizar
Buscar en ésta Colección
Ryan Star, supernova
Escuchar en Google |
Destruir
WeMightFall, Ryan Star
Escuchar en Google |
Destruir
Transfiriendo datos desde iussu.ath.cx...

- En la Colección Otros usted puede modificar el nombre, los tags y la imagen.

TAGS

Estos permiten agilizar la búsqueda por parte del usuario o de los diferentes usuarios.
Por eso es necesario que agregue los tags a los diferentes archivos.

20. Ingrese los tags en la parte superior del archivo, coloque el cursor sobre la frase "clic para añadir tags", e ingrese los tags que usted desee al archivo, separados por comas.

21. Haga clic sobre el botón guardar tags, para que queden con el archivo, podrá verlos al lado izquierdo en el cuadro "Buscar en esta colección",
Nota: si utiliza con frecuencia ciertos tags, estos cambian en color y tamaño.

Cómo por ejemplo en la imagen a continuación.

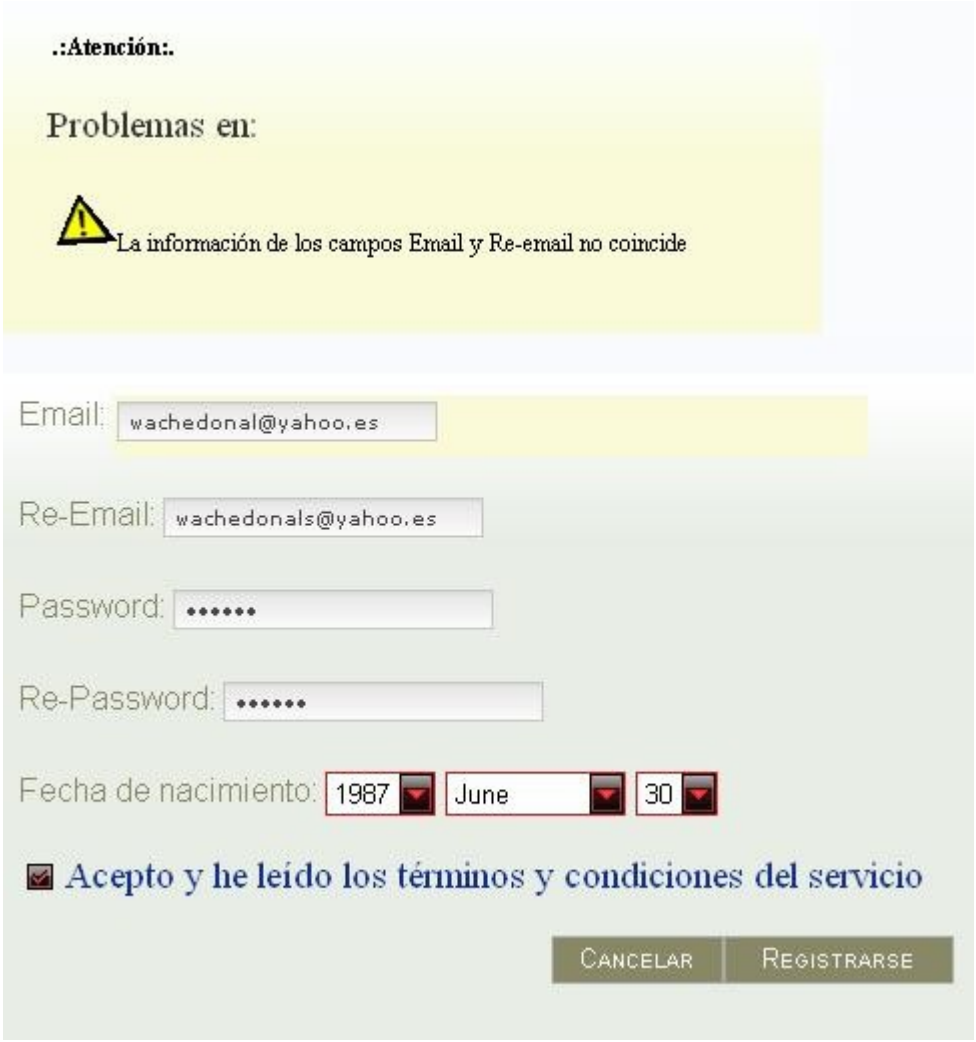
MENSAJES DE IUSSU

Los mensajes en IUSSU se presentan en la parte superior, en color amarillo.

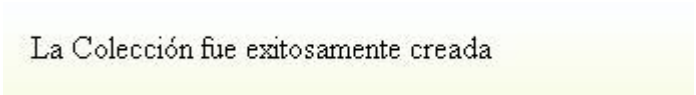
Algunos mensajes son de error o de información no valida así como otros mensajes son informativos.

Es decir cuando se ingresa la información no valida en los campos del formulario de registro, el campo es resaltado en un color amarillo e inmediatamente aparece el mensaje en la parte superior. Ejemplo

Nuestro usuario Wachedonal, se esta registrando, pero digita erróneamente el correo, como lo ilustra la imagen, aparece un mensaje de advertencia.



También existe otra clase de mensajes como por ejemplo cuando se crea una colección.



ANEXO G

MANUAL DEL SISTEMA

El manual del sistema fue generado automáticamente con Rdoc, es una colección de páginas HTML y puede consultarse en la carpeta doc de la aplicación, o en la cuenta de administrador iussu.service en la parte superior derecha bajo el link Manual del Sistema.